

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

Приладобудівний факультет

Кафедра приладобудування

До захисту допущено:

Завідувач кафедри

\_\_\_\_\_ Юрій КИРИЧУК

«\_\_» \_\_\_\_\_ 20\_\_ р.

**Дипломний проєкт**

**на здобуття ступеня бакалавра**

**за освітньо-професійною програмою «Комп'ютерно-інтегровані технології  
та системи точної механіки»**

**спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»**

**на тему: «Смарт-система приготування кави»**

Виконав:

студент IV курсу, групи ПМ-61

Олійник Владислав Сергійович \_\_\_\_\_

Керівник:

к.т.н. , доцент

Гришанова Ірина Аркадіївна \_\_\_\_\_

Консультант з назва розділу:

д.т.н., професор

Антонюк Віктор Степанович \_\_\_\_\_

Рецензент:

к.т.н., доцент

Шевченко Вадим Володимирович \_\_\_\_\_

Засвідчую, що у цьому дипломному  
проєкті немає запозичень з праць інших  
авторів без відповідних посилань.

Студент \_\_\_\_\_

Київ – 2020 року

## ВІДОМІСТЬ ДИПЛОМНОГО ПРОЄКТУ

[illegible]

				ДП ПМ61 15.000.00		
	ПБ	Підп.	Дата	Відомість дипломного проекту	Лист	Листів
Розробн.	Олійник В. С.				1	
Керівн.	Гришанова І. А.				КПІ ім. Ігоря Сікорського Каф. ПБ Гр. ПМ-61	
Консульт.	Антонюк В. С.					
Н/контр.						
Зав.каф.	Киричук Ю. В.					

**Пояснювальна записка  
до дипломного проєкту  
на тему: «Смарт-система приготування кави»**

Київ – 2020 року  
Національний технічний університет України  
«Київський політехнічний інститут  
імені Ігоря Сікорського»

Факультет \_\_\_\_\_ Приладобудівний \_\_\_\_\_  
(повна назва)

Кафедра \_\_\_\_\_ Приладобудування \_\_\_\_\_  
(повна назва)

Рівень вищої освіти – перший (бакалаврський)

Спеціальність (спеціалізація) 151 Автоматизація та комп'ютерно-інтегровані технології (Комп'ютерно-інтегровані технології та системи точної механіки) \_\_\_\_\_  
(код і назва)

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
\_\_\_\_\_ Юрій КИРИЧУК.  
(підпис) (ініціали, прізвище)  
« \_\_\_\_ » \_\_\_\_\_ 2020 р.

**ЗАВДАННЯ**  
**на дипломний проєкт студенту**

Олійнику Владиславу Сергійовичу

\_\_\_\_\_  
(прізвище, ім'я, по батькові)

**1. Тема проєкту**

Смарт-система приготування кави

Керівник проєкту Гришанова Ірина Аркадіївна, к.т.н., доцент \_\_\_\_\_  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «25» травня 2020 р. №1180-с

**2. Строк подання студентом проєкту** \_\_\_\_\_

**3. Вихідні дані до проєкту:** точність термодатчиків - 0.25 градуси Цельсія, точність тензодатчиків - 1 грам, діапазон тиску - 0 – 1.5 бар, напруга живлення - 220 В

\_\_\_\_\_  
\_\_\_\_\_

#### 4. Зміст розрахунково-пояснювальної записки

Перелік скорочень. Вступ. 1. Проектно-конструкторський розділ. 1.1. Огляд і аналіз існуючих смарт-систем, що застосовуються для приготування кави. 1.2. Вибір датчиків, компонентів та платформи управління. 1.3. Програмне забезпечення. 1.4. Прототип смарт-кавоварки. 2. Технологічний розділ. 2.1. Короткий опис складальної одиниці. 2.2. Визначення основних показників технологічності. 2.3. Визначення додаткових показників технологічності. 2.4. Визначення комплексного показника технологічності виробу. 2.5. Схема складального складу. 2.6. Технологічна схема складання. 2.7. Розмірний ланцюг. Висновки. Перелік посилань. Додатки.

#### 5. Перелік (ілюстративного) графічного матеріалу (з точним зазначенням обов'язкових креслеників, плакатів, тощо)

5.1 Схема підключення електронних компонентів до Raspberry Pi – 1 арк. фА1

5.2 Схема взаємодії компонентів смарт-системи приготування кави – 1 арк. фА1

5.3 Принципова електрична схема смарт-системи приготування кави – 1 арк. фА1

5.4 3D модель прототипу смарт-системи приготування кави – 1 арк. фА1

#### 6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Технологічний	Антонюк В. С.		

#### 7. Дата видачі завдання 11.03.2020 р.

#### Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Строк виконання етапів проекту	Примітка
1.	Огляд і аналіз аналогів	11.03-21.03.2020р.	
2.	Розробка схем смарт-системи	22.03-01.04.2020р.	
3.	Вибір датчиків і модулів	01.04-10.04.2020р.	
4.	Розробка програмного забезпечення	11.04-15.04.2020р.	
5.	Розробка макетного стенду	15.04-25.04.2020р.	
6.	Розробка прототипу смарт-системи	26.04-04.05.2020р.	
7.	Розробка КД	05.05-15.05.2020р.	
8.	Виконання технологічного розділу	16.05-22.05.2020р.	
9.	Підготовка ПЗ	22.05-01.06.2020р.	

Студент

Керівник проекту

Олійник В.С.

Гришанова І.А.

## АНОТАЦІЯ

Загальний обсяг роботи 65 сторінок із них: 11 сторінок додатків. Також включає в себе 7 таблиць, 10 посилань, 36 рисунків (схеми, фото та ін.).

Даний проєкт присвячений створенню професійної кавомашини для подальшого використання її у закладах громадського харчування. Вибір теми обґрунтований тим, що сучасні професійні еспресо машини коштують дорого, мають складну конструкцію, що ускладнює заміну деталей. Також кавоварки не виробляються в Україні.

Метою роботи є створення робочого прототипу кавомашини з використанням серійних вузлів існуючих кавоварок. Також планується значно здешевити затрати на виробництво кавових машин та в подальшому налагодити виробництво в Україні.

Ключові слова: Raspberry Pi, Python, смарт-кавоварка, інтерфейс, програмне забезпечення, датчик.

## ABSTRACT

The total volume of work is 65 pages of them: 11 pages of appendices. Also includes 7 tables, 10 links, 36 figures (diagrams, photos, etc.).

This project is dedicated to the creation of a professional coffee machine for further use in catering establishments. The choice of theme is justified by the fact that modern professional espresso machines are expensive, have a complex design that makes it difficult to replace parts. Also, coffee machines are not produced in Ukraine.

The aim of the work is to create a working prototype of a coffee machine using serial units of existing coffee machines. It is also planned to significantly reduce the cost of production of coffee machines and further establish production in Ukraine.

Keywords: Raspberry Pi, Python, smart coffee maker, interface, software, sensor.

## ЗМІСТ

Перелік скорочень .....	9
Вступ.....	10
1. Проектно-конструкторський розділ.....	12
1.1. Огляд і аналіз існуючих смарт-систем, що застосовуються для приготування кави.....	12
1.2. Вибір датчиків, компонентів та платформи управління.....	14
1.2.1. Raspberry Pi.....	14
1.2.2. Датчик ваги (тензодатчик) .....	16
1.2.3. Чотирьох модульне реле.....	20
1.2.4. Термопара К-типу.....	21
1.2.5. Датчик тиску.....	23
1.2.6. Дисплей сенсорний.....	26
1.2.7. Структурна схема.....	27
1.3. Програмне забезпечення.....	28
1.3.1. Використані технології.....	28
1.3.2. Інтерфейс управління смарт-системою приготування кави.....	29
1.4. Прототип смарт-кавоварки.....	33
1.4.1. Опис конструкції та принцип дії.....	33
1.4.2. Складові компоненти прототипу.....	36
2. Технологічний розділ.....	41
2.1. Короткий опис складальної одиниці.....	41
2.2. Визначення основних показників технологічності.....	42
2.3. Визначення додаткових показників технологічності.....	43
2.4. Визначення комплексного показника технологічності виробу.....	44
2.5. Схема складального складу.....	45
2.6. Технологічна схема складання.....	47
2.7. Розмірний ланцюг.....	49
Висновки.....	52
Перелік посилань.....	53
Додаток А.....	54

					ДП ПМ6115.000.000 ПЗ					
Зм.	Арк.	№ докум.	Підпис	Дата						
Розроб.	Олійник				Смарт-система приготування кави			Літера	Аркуш	Акривіш
Перевір.									8	64
Реценз.										
Н. Контр.										
Затверд.	Гришанова							ПБФ, 4 курс, ПМ-61		



## ПЕРЕЛІК СКОРОЧЕНЬ

GPIO – General Purpose Input Output

LAN – Local Area Network

USB – Universal Serial Bus

HDMI – High Definition Multimedia Interface

VGA – Video Graphics Array

TFT – Thin Film Transistor

АЦП – Аналого-цифровий перетворювач

ОС – Операційна система

ТЕН - Трубчастий електронагрівник

					ДП ПМ6115.000.000 ПЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

## ВСТУП

Проект виробництва еспreso-машини для професійного використання в закладах громадського харчування.

Наразі ринок споживання кави в українських кафе та кав'ярнях оцінюється в більш ніж 500 млн євро. За даними [1] Асоціації Спешелті Кави (SCA) на українському кавовому ринку наразі функціонує більше ніж 11500 кав'ярень. Згідно даних SCA кількість кав'ярень починаючи з 2010 р. збільшилась на 140% та продовжує зростати.

Еспreso кавоварка – один з ключових елементів сучасної кав'ярні. Зараз основними виробниками кавових машин для професійного використання є Італія, Іспанія та США. Також у 2019 році на ринок вийшли компанії виробники обладнання з Австралії та Росії.

Більшість основних елементів кавоварки може бути виготовлене на території України. Процес фінальної збірки також може бути налагоджений. При цьому вартість виготовлення буде значно нижчою за аналогічні продукти закордонного виробництва. На даному етапі планується досягнути 40-60% зниження вартості порівняно з аналогічними за функціоналом кавомашинами іноземного виробництва.

Реалізація проекту планується в наступні етапи:

- 1) створення робочого прототипу з використанням серійних вузлів існуючих кавоварок. На цьому етапі підбирається компонентна база електроніки, створюється програмний код, ведеться робота над зовнішнім та внутрішнім дизайном машини. Основні критерії успіху на цьому етапі створити функціональний прототип, у якому за допомогою програмного забезпечення можна буде контролювати кожен з процесів приготування кави. Також досягти легкості встановлення та обслуговування кавоварки, зручності у роботі з нею. Управління на прототипі здійснюватиметься завдяки мікроконтролеру Raspberry Pi.

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		10

- 2) Визначення вузлів та компонентів виготовлення яких можливе в Україні.  
Наразі 90% елементів механічної частини кавоварки можуть бути повністю виготовлені в Україні (рама та корпус, бойлер, крани, з'єднувальні трубки). 100% електричних та електронних компонентів може бути зібрана та запрограмована на базі наявних ринкових рішень (Omron, Siemens та ін.).
- 3) Запуск дрібної серії машин в виробництво.
- 4) Виведення продукту на міжнародний ринок (виставлення перших серійних зразків обладнання на виставках HOST та WOC у 2021 році.

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

## 1. ПРОЄКТНО-КОНСТРУКТОРСЬКИЙ РОЗДІЛ

### 1.1. Огляд і аналіз існуючих смарт-систем, що застосовуються для приготування кави.

Кавоварки або кавомашини – це прилади для приготування кави. Існує багато різних типів кавоварок, що мають різні принципи заварювання. У кав'ярнях, ресторанах та кафе використовують професійні кавомашини, або як їх ще називають еспресо машини, так як потрібно готувати велику кількість гарячих напоїв. Принцип приготування кави у таких кавоварок простий – кипляча вода під тиском проходить через фільтр із меленою кавою.

Сьогодні на ринку представлено багато професійних еспресо машин різних типів та цінових сегментів. На жаль, в Україні немає виробництва кавомашин, але наприклад у Італії ця індустрія дуже розвинена.

Для огляду було вибрано дві еспресо машини від італійських виробників.



Рис. 1.1. Victoria Arduino Adonis

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12



Рис. 1.2. Nuova Simonelli Appia Life

Victoria Arduino Adonis – професійна кавова машина, що обладнана LCD дисплеєм, має 2 групи заварювання кави та сучасний дизайн. Ціна Victoria Arduino Adonis складає \$11 000.

Nuova Simonelli Appia Life – дешевший аналог Victoria Arduino Adonis, що також обладнаний LCD дисплеєм. Ціна цієї кавомашини – \$7 500.

Одним із недоліків цих кавоварок є надто висока ціна. Для малого бізнесу це значні ціни, тому вони змушені купувати кавомашини які вже були у використанні. Проєкт смарт-системи приготування кави передбачає здешевлення виробництва кавових машин за рахунок використання у якості контролера доступний одноплатний комп'ютер Raspberry Pi.

Керування приведеними вище кавоварками відбувається фізичними кнопками і деяка інформація виводиться на невеличкий дисплей. Використання фізичних кнопок для управління кавоваркою не є сучасним та зручним рішенням. Наприклад для додавання нового функціоналу потрібно змінювати дизайн самої кавоварки для розташування елементів керування. Також недоліком є малий дисплей так як на ньому не зручно переглядати інформацію про поточний стан кавової машини. Проєкт смарт-системи приготування кави враховує ці недоліки. Керування смарт-системою відбувається на сенсорному дисплеї. Для додавання нових функцій не потрібно змінювати дизайн самої

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

машини, а лише зробити зміни у програмному інтерфейсі, що значно зручніше, швидше та дешевше.

Модульна конструкція – це суттєва перевага будь якого приладу. Адже легка заміна частин у випадку виходу з ладу певних вузлів робить обслуговування значно дешевшим.

Можна зробити висновок, що основними перевагами смарт-системи приготування кави, що описана у даному проєкті, є:

- керування смарт-кавоваркою відбувається на великому та зручному сенсорному екрані
- модульна конструкція (легка заміна частин у випадку виходу з ладу певних вузлів)
- можливість більш точного контролю параметрів приготування напоїв
- низькі затрати на виробництво – як наслідок конкурентна ціна

## **1.2. Вибір датчиків, компонентів та платформи управління**

### **1.2.1. Raspberry Pi**

Для управління електронними компонентами був вибраний одноплатний комп'ютер Raspberry Pi.

Raspberry Pi – потужний, сучасний, одноплатний комп'ютер. Цей комп'ютер користується популярністю у всьому світі. Для програмування на ньому можна використовувати майже всі популярні мови програмування, але найпоширенішими є: Python, C, C++, Java.

Використанням Raspberry Pi можна вирішити більшість задач. Комп'ютер застосовують для здобуття навичок програмування, для автоматизації різноманітних процесів, створення незвичайних проєктів та навіть для вирішення задач у промисловості.

Працює під управлінням Linux, а також пропонує набір виходів GPIO через які можна керувати електронними компонентами. Всього є 40 GPIO виходів.

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		14

Серед них: два контакти 5 В та два 3.3 В, а також 8 контактів заземлення. Всі інші виходи є виходами загального призначення.



Рис. 1.3. Raspberry Pi 4 B

Табл. 1.1. Характеристики Raspberry Pi 4 B

Процесор	1,5 ГГц 64-бітний чотирьохядерний процесор ARM Cortex-A72
Об'єм оперативної пам'яті	1 ГБ LPDDR4 SDRAM
Мережеві інтерфейси	<ul style="list-style-type: none"> <li>• 2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless</li> <li>• LAN</li> <li>• Bluetooth 5.0</li> <li>• BLE</li> <li>• Gigabit Ethernet</li> </ul>
USB порти	<ul style="list-style-type: none"> <li>• 2 × USB 3.0</li> <li>• 2 × USB 2.0</li> </ul>

Відеовихід	Два micro-HDMI з підтримкою двох моніторів з роздільною здатністю до 4K
Графічний інтерфейс	VideoCore VI, що підтримує OpenGL ES 3.x
Роз'єм живлення	USB-C
Напруга живлення	5В

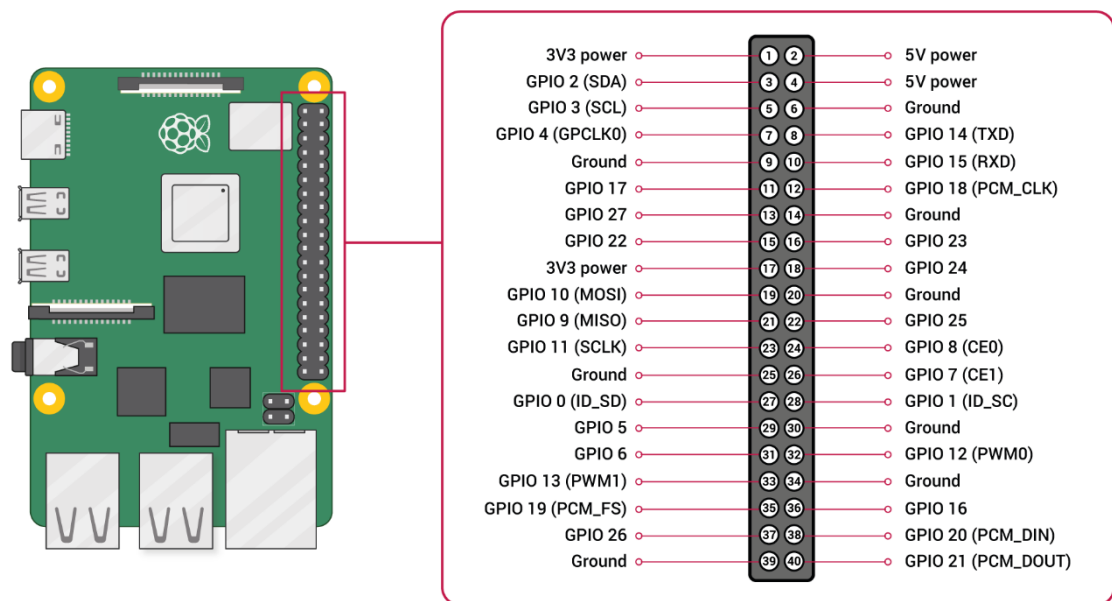


Рис. 1.4 GPIO виходи [2]

### 1.2.2. Датчик ваги (тензодатчик)

Тензодатчик – датчик який перетворює величину деформації у електричний сигнал. У своїй конструкції має плівкові резистори, які змінюють свій опір при деформації. Використовується разом із АЦП НХ711 який фіксує зміну значень резисторів. Датчик вбудований у конструкцію підставки для вимірювання ваги на групах заварювання (рис. 1.5).



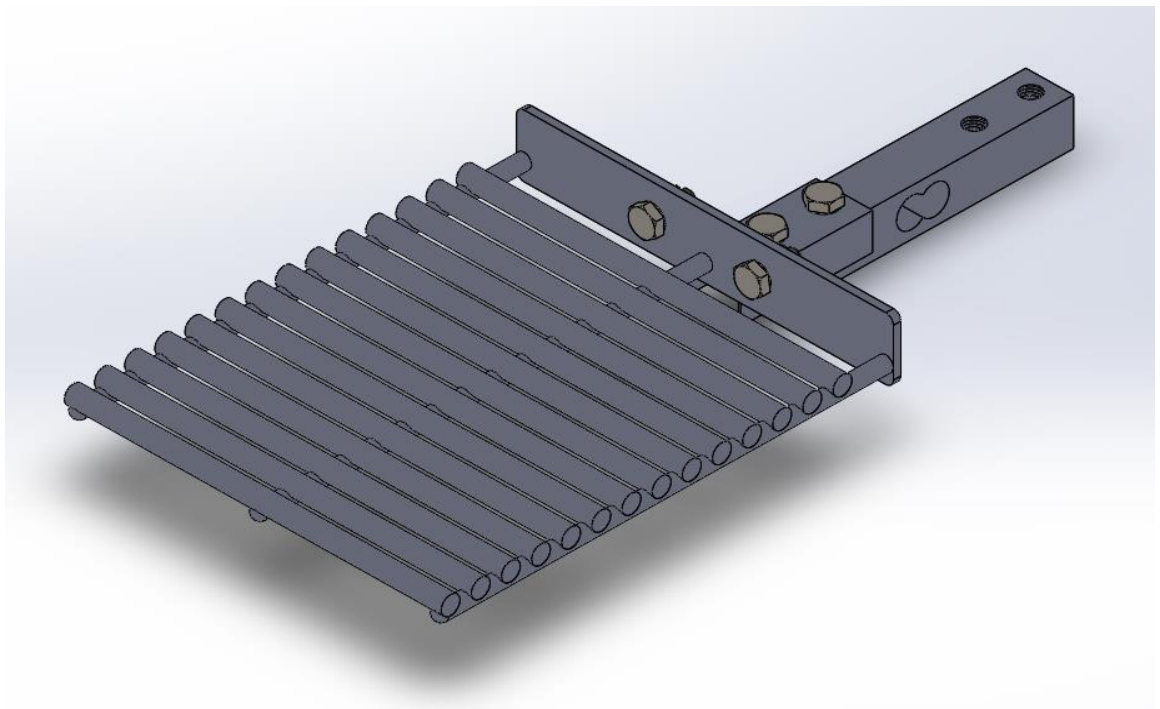


Рис. 1.5. Конструкція підставки для чашки на групі заварювання із можливістю виміру ваги

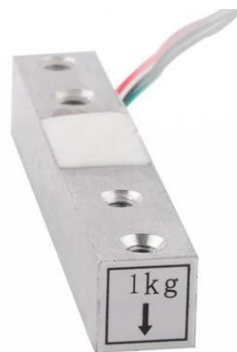


Рис. 1.6. Датчик ваги (тензодатчик)

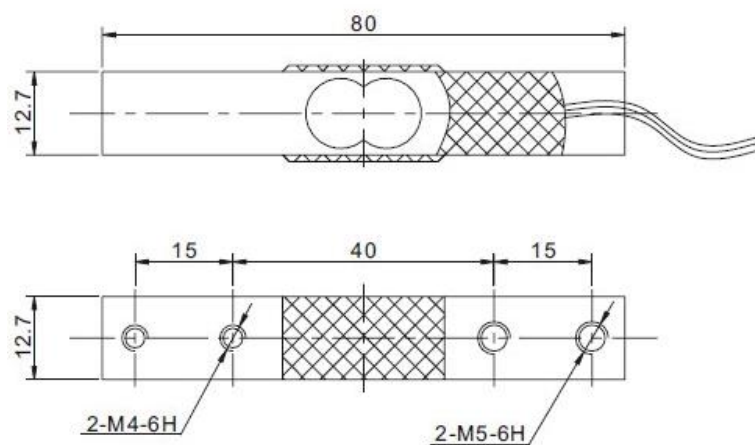


Рис. 1.7. Ескіз датчика [3]

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

Табл. 1.2. Характеристики датчика ваги

Розміри	80 x 12.7 x 12.7 мм
Матеріал	алюміній
Комплексна похибка	0,2 % FS
Імпеданс	$1000 \pm 50\Omega$
Рекомендована напруга	5 – 10 В
Максимальна вага	до 1 кг

HX711 - це точний 24-розрядний аналого-цифровий перетворювач (АЦП), призначений для взаємодії безпосередньо із мостом тензодатчика.

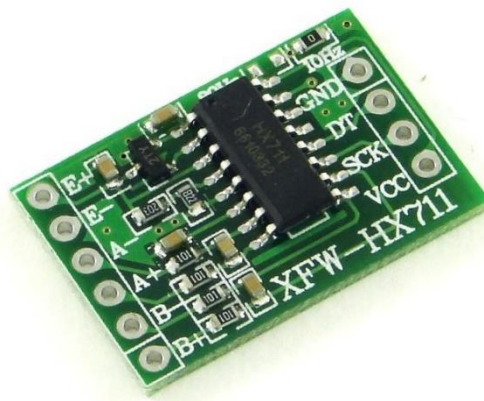


Рис. 1.8. HX711 Модуль датчиків ваги

АЦП має два роз'єми:

- J1 роз'єм:
  - E+, E- живлення тензодатчика;
  - A-, A+ диференціальний вхід каналу A;
  - B-, B+ диференціальний вхід каналу B;
- J2 роз'єм:
  - VCC – напруга живлення;
  - GND – заземлення;
  - DT, SCK – інформаційні лінії;

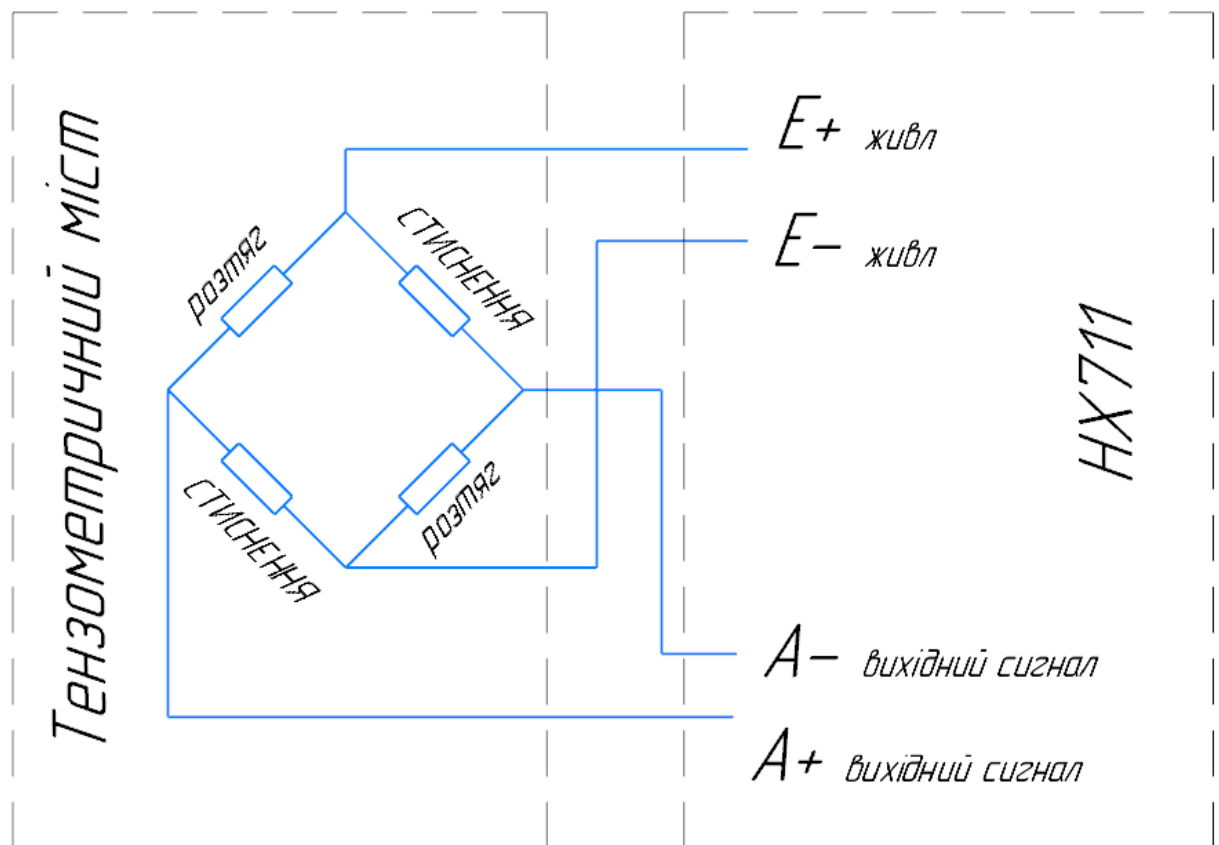


Рис. 1.9. Схема підключення мостового датчика до АЦП HX711

Для взаємодії із тензодатчиком використовується бібліотека із відкритим вихідним кодом [4]. Для початку роботи потрібно відкалібрувати ваги. Для цього потрібно дізнатись еталонне значення. Потрібен предмет із відомою вагою або гиря для вагів. Цей предмет потрібно помістити на датчик та зняти значення у програмі. Спочатку запускається наступний програмний код:

```
from hx711 import HX711
```

```
hx = HX711(5, 6)
```

```
hx.set_reference_unit(1)
```

```
hx.reset()
```

```
hx.tare()
```

```
while True:
```

```

val = hx.get_weight(5)
print(val)
hx.power_down()
hx.power_up()
time.sleep(0.1)

```

При зважуванні 100 грамової гири було отримано значення 193 000. Для розрахунку еталонного значення потрібно отримане значення поділити на вагу гири. Розрахунок:  $193\ 000 / 100 = 1930$  – це еталонне значення. Його потрібно передати у функцію `set_reference_unit(1930)`.

### 1.2.3. Чотирьох модульне реле

Має 4 комутуючих реле (рис. 1.10). Кожному реле потрібно 15-20мА для спрацьовування і напруга живлення модуля 5 Вольт. Розміри 4.9 см x 5.1 см x 1.8 см.



Рис. 1.10. 4-х канальний модуль реле

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		20

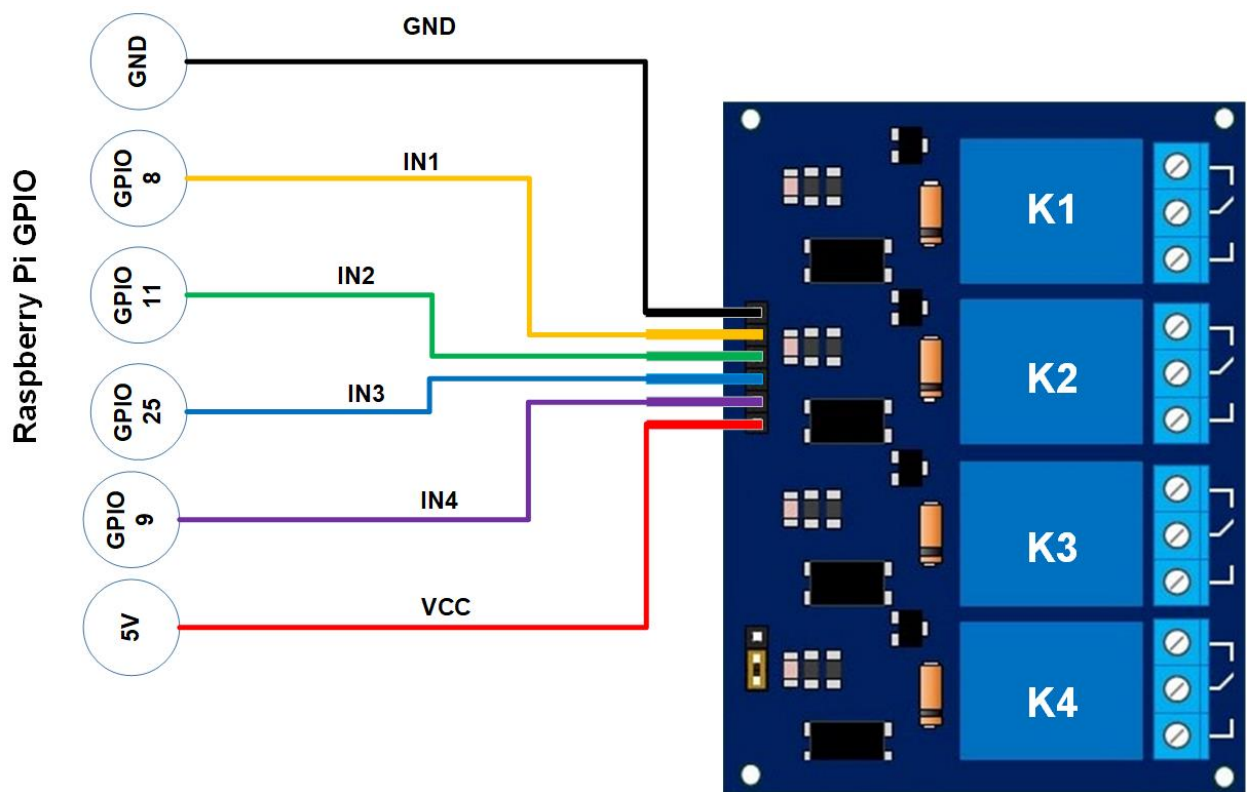


Рис. 1.11. Схема підключення 4-х каналного реле

#### 1.2.4. Термопара К-типу

Термопара – електричний пристрій, який складається із двох різнорідних електричних провідників, що утворюють електричний перехід. Термопара створює залежну від температури напругу у результаті термоелектричного ефекту і цю напругу можна інтерпретувати для вимірювання температури

Тип К (хромель - алюмель) є найбільш поширеною термопарою загального призначення із чутливістю  $41 \text{ мкВ} / ^\circ\text{C}$ .

Використовується разом із цифровим підсилювачем MAX6675.

MAX6675 виконує компенсацію холодного з'єднання і оцифровує сигнал від термопари типу К. Перш ніж перетворити термоелектричні напруги в еквівалентні температурні значення, необхідно компенсувати різницю між стороною холодного з'єднання термопари (температура навколишнього середовища MAX6675) і віртуальною орієнтацією  $0^\circ\text{C}$ . Для термопари типу К напруга змінюється на  $41 \text{ мкВ} / ^\circ\text{C}$ , що приводить характеристику термопари до наступного лінійного рівняння:

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

$$V_{OUT} = (41 \text{ мкВ} / ^\circ \text{C}) * (T_R - T_{AMB}) , \text{ де} \quad (1.1)$$

$V_{OUT}$  - вихідна напруга термопари (мкВ).

$T_R$  - температура віддаленого з'єднання термопари ( $^\circ \text{C}$ ).

$T_{AMB}$  - температура навколишнього середовища ( $^\circ \text{C}$ ).

Діапазон вимірювальних температур дозволяє спокійно працювати з температурою у бойлерах.

Табл. 1.3. Характеристики MAX6675

Робоча напруга	3.0 ~ 5.5 В
Тип термопари	К-тип
Робочий температурний діапазон	0 – 700 $^\circ \text{C}$
Температурна похибка	0.25 $^\circ \text{C}$



Рис. 1.12. Термопара К-типу із цифровим підсилювачем MAX6675 [5]

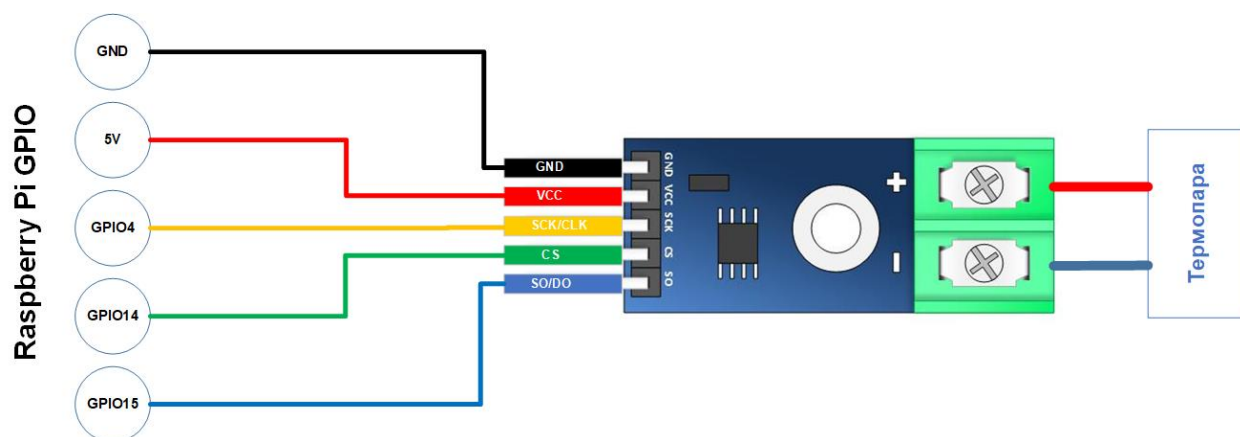


Рис. 1.13. Схема підключення

### 1.2.5. Датчик тиску



Рис. 1.14. Датчик тиску

Табл. 1.4. Характеристики датчика тиску

Робоча напруга	5В
Вихідна напруга	0.5 - 4.5V
Матеріал	Сталь
Робочий діапазон тиску	0 - 1.2 Мпа / 12 бар / 175 psi
Робоча температура, діапазон	0 – 85 ° C
Похибка вимірювання	± 3%
Призначення:	неагресивні газу і рідини

Датчик тиску (рис. 1.14) використовується у паровому бойлері. Діапазон тиску який має підтримуватись у бойлері від 1 до 1.5 бар. Цей датчик повністю



підходить для кавоварки, так як робочий діапазон тиску у датчика – 0-12 бар. Разом із цим датчиком використовується аналогово-цифровий перетворювач ADS1015 (рис. 1.15).



Рис. 1.15. Перетворювач ADS1015

Табл. 1.5. Характеристики ADS1015

Інтерфейс	I2C
Напруга живлення	від 2 В до 5 В
Струм живлення у безперервному режимі	150 мкА
Розрядність АЦП	12
Кількість каналів	4 або 2 дифференційних
Діапазон вхідних напруг	$\pm 256\text{ мВ}$ до $\pm 6.144\text{ В}$
Частота оцифровки	от 128 до 3300 відліків в секунду



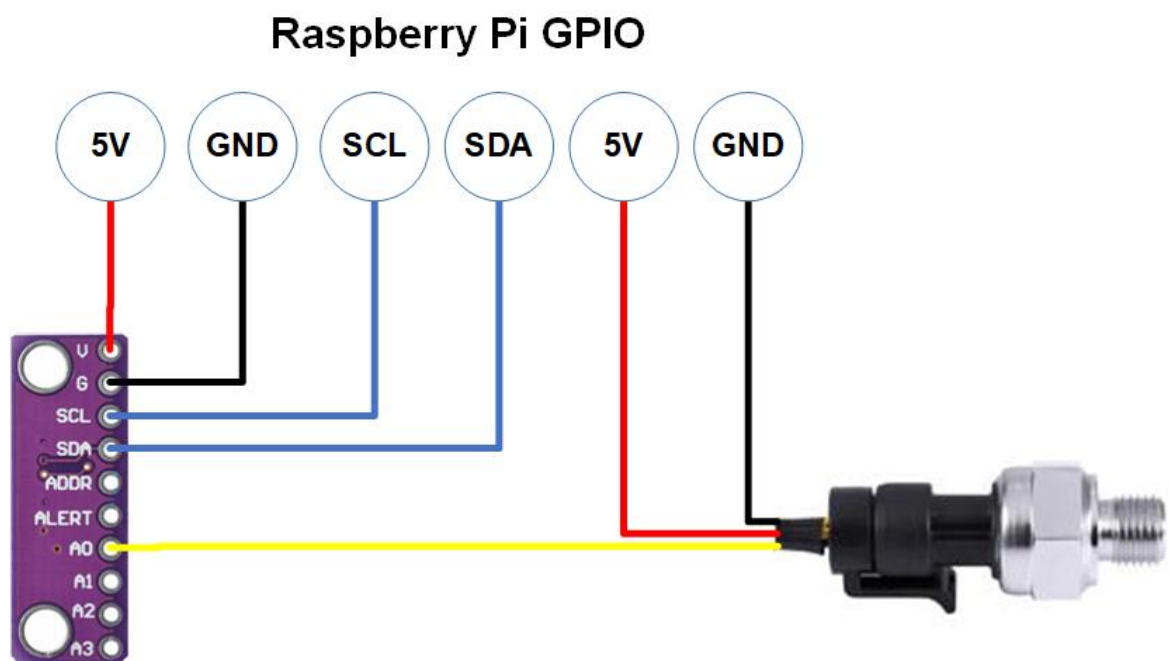


Рис. 16. Схема підключення датчика тиску та ADS1015

Для отримання коректних значень тиску у потрібних одиницях (у даному випадку результатом буде тиск у барах) потрібно обрахувати за формулою:

$$P = (x - 4400) * 5,5 * 10^{-4}, \text{ де} \quad (1.2)$$

$x$  – значення отримане із датчику тиску

Ця формула врахована при розробці програмного забезпечення, тому користувачу відразу виводиться значення у одиницях виміру тиску – барах.

### 1.2.6. Дисплей сенсорний



Рис. 1.17. Дисплей сенсорний [6]

П'яти дюймовий дисплей із ємнісним сенсором. Призначений для роботи із комп'ютерами які мають HDMI інтерфейс. Використовується для керування смарт-кавоваркою.



Рис. 1.18. Сенсорний дисплей кавоварки із інтерфейсом

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

Табл. 1.6. Характеристики сенсорного дисплея

Тип матриці	TFT
Роздільна здатність	800x480 (підтримується масштабування до 1920x1080)
Сумісність	Підтримує популярні міні-ПК, такі як Raspberry Pi, BB Black, Banana Pi, а також звичайні настільні комп'ютери
Звуковий вихід	3,5-мм аудіороз'єм Роз'єм динаміка
Відеовходи	HDMI VGA

### 1.2.7. Структурна схема

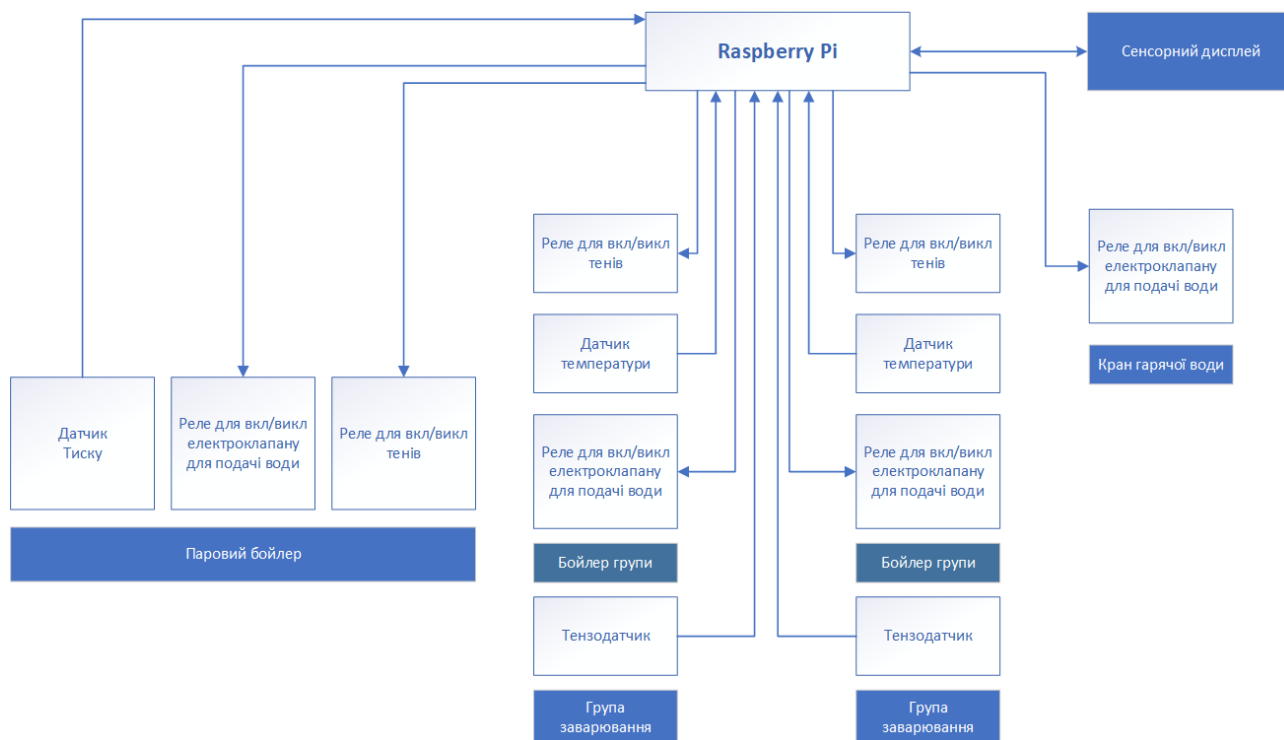


Рис. 1.19. Структурна схема взаємодії Raspberry Pi із електронними компонентами

Стрілки що ведуть до Raspberry Pi означають, що плата отримує від датчиків дані. Стрілки що ведуться від одноплатного комп'ютера – це Raspberry Pi посилає сигнал у компоненти. Дисплей працює у дві сторони.

### 1.3. Програмне забезпечення

#### 1.3.1. Використані технології

Програмне забезпечення для смарт-кавоварки розроблено на мові програмування Python. Взаємодія із електронними компонентами відбувається за допомогою цифрових портів GPIO Raspberry Pi. Для управління портами була використана бібліотека із відкритим вихідним кодом RPi.GPIO. Для створення інтерфейсу користувача був використаний Kivy фреймворк.

Python – інтерпретована, високорівнева мова програмування зі строгою динамічною типізацією. Python підтримує такі парадигми програмування: процедурне, функціональне, об'єктно-орієнтоване та аспектно-орієнтоване. Підтримується модульність, тобто розбиття програм на модулі та пакети модулів. Серед основних її переваг можна назвати такі:

- чистий синтаксис (для виділення блоків слід використовувати відступи);
- переносність програм (що властиве більшості інтерпретованих мов);
- стандартний дистрибутив має велику кількість корисних модулів (включно з модулем для розробки графічного інтерфейсу);
- можливість використання Python в діалоговому режимі (дуже корисне для експериментування та розв'язання простих задач);
- стандартний дистрибутив має просте, але разом із тим досить потужне середовище розробки, яке зветься IDLE і яке написане мовою Python;
- зручний для розв'язання математичних проблем (має засоби роботи з комплексними числами, може оперувати з цілими числами довільної величини, у діалоговому режимі може використовуватися як потужний калькулятор);
- відкритий код (можливість редагувати його іншими користувачами).

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		28

Python має ефективні структури даних високого рівня та простий, але ефективний підхід до об'єктно-орієнтованого програмування. Елегантний синтаксис Python, динамічна обробка типів, а також те, що це інтерпретована мова, роблять її ідеальною для написання скриптів та швидкої розробки прикладних програм у багатьох галузях на більшості платформ [7].

Kivy – це безкоштовна бібліотека Python із відкритим вихідним кодом для розробки мобільних додатків та іншого програмного забезпечення. Бібліотека поширюється за умовами ліцензії MIT і може працювати на Android, iOS, Linux та Windows.

Фреймворк містить такі елементи для створення програм:

- широка підтримка вводу для подій, що стосуються мультитачів, характерних для миші, клавіатури, TUIO та ОС,
- графічна бібліотека, що використовує лише OpenGL ES 2 та заснована на Vertex Buffer і шейдерах,
- широкий спектр віджетів, які підтримують мультитач,
- проміжну мову (Kv), яка використовується для легкого проєктування користувацьких віджетів.

### **1.3.2. Інтерфейс управління смарт-системою приготування кави**

Для розробки інтерфейсу було використана бібліотека KivyMD. KivyMD – це сукупність віджетів, сумісних з Material Design, для використання з Kivy, основою для крос-платформних, сенсорних графічних програм. [8].

KivyMD максимально наближає дизайн програм до специфікації Google Material Design, не приносячи шкоди простоті використанню або продуктивності програми.

Управління системою приготування кави відбувається через інтерфейс який складається із двох вкладок: Main (Головна) та Settings (Налаштування).

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		29



Рис. 1.20. Інтерфейс управління кавоваркою (вкладка Main)

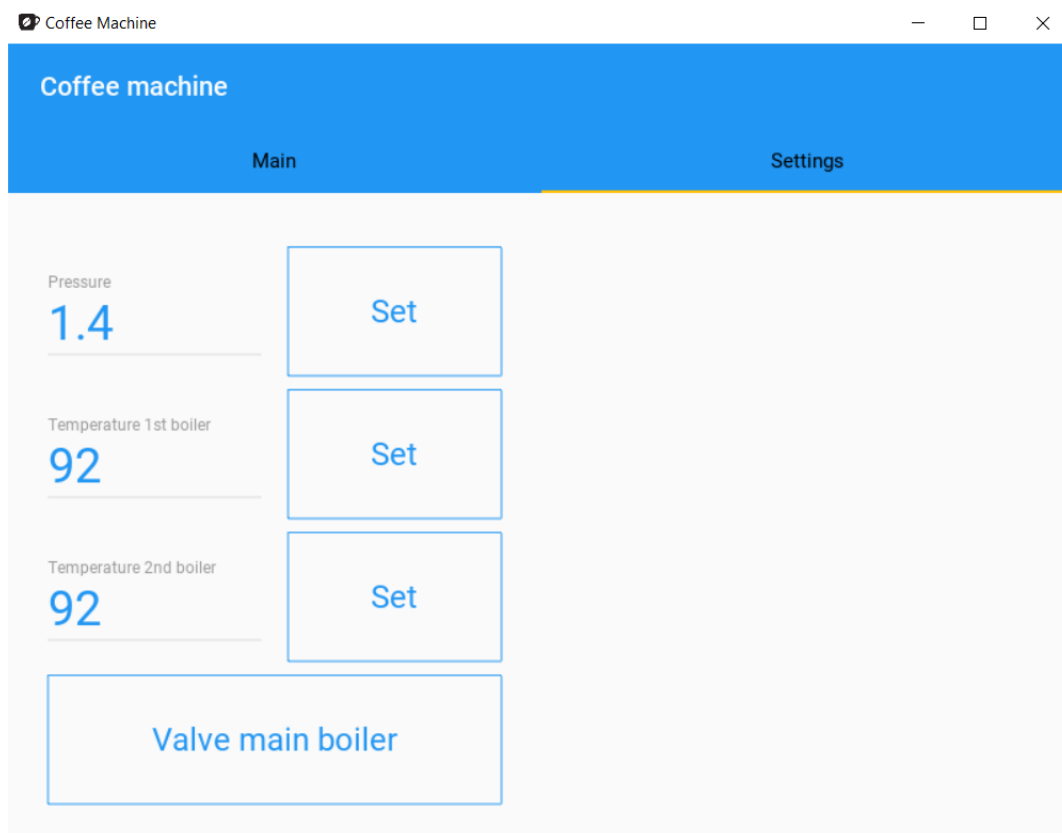


Рис. 1.21. Інтерфейс управління кавоваркою (вкладка Settings)

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30

Вкладка Main (рис. 1.20) включає в себе багато компонентів, зокрема тут відображена основна інформація про стан кавоварки, такі як: температура у бойлерах на групах заварювання та значення тиску у паровому бойлері. У вкладці Main також присутні кнопки управління. Кнопка «Hot water» при натисканні вмикає реле і струм подається на електроклапан, що відкриває кран гарячої води. Великі кнопки, що розташовані по центру дисплея, відповідають за подачу води у групи заварювання. На самих кнопках відображена інформація про вагу чашки, що знаходиться на вагах групи. При натисканні на кнопку запускається таймер та вмикається реле і струм подається на електроклапан, який в свою чергу дозволяє подачу води на групу. Разом із цим вмикається реле, що подає струм на мотор подачі води (помпу) і вмикає його. При повторному натисканні реле вимикаються, таймер зупиняється.

Вкладка Settings (рис. 1.21) складається із полів вводу та кнопок. Біля кожного поля знаходиться кнопка «Set», яка підтверджує введені значення. В цих полях вводу можна задати бажану температуру у бойлерах груп заварювання та тиск у паровому бойлері. Але є обмеження: не можна встановити температуру більшу ніж 96 °C та значення тиску більше ніж 1.5 бар. Це зроблено для запобігання перегрівання бойлерів. Програма буде тримати задану температуру та тиск протягом усього часу роботи із системою. Кнопка «Valve main boiler» подає струм на електроклапан парового бойлеру та на помпу, яка починає закачувати воду у паровий бойлер. Щоб вимкнути подачу струму, і цим самим зупинити закачку води, потрібно повторно натиснути на кнопку.

Для підтримування заданих температур у бойлерах груп заварювання та заданого значення тиску у паровому бойлері, розроблено алгоритм представлений на рис. 1.22.

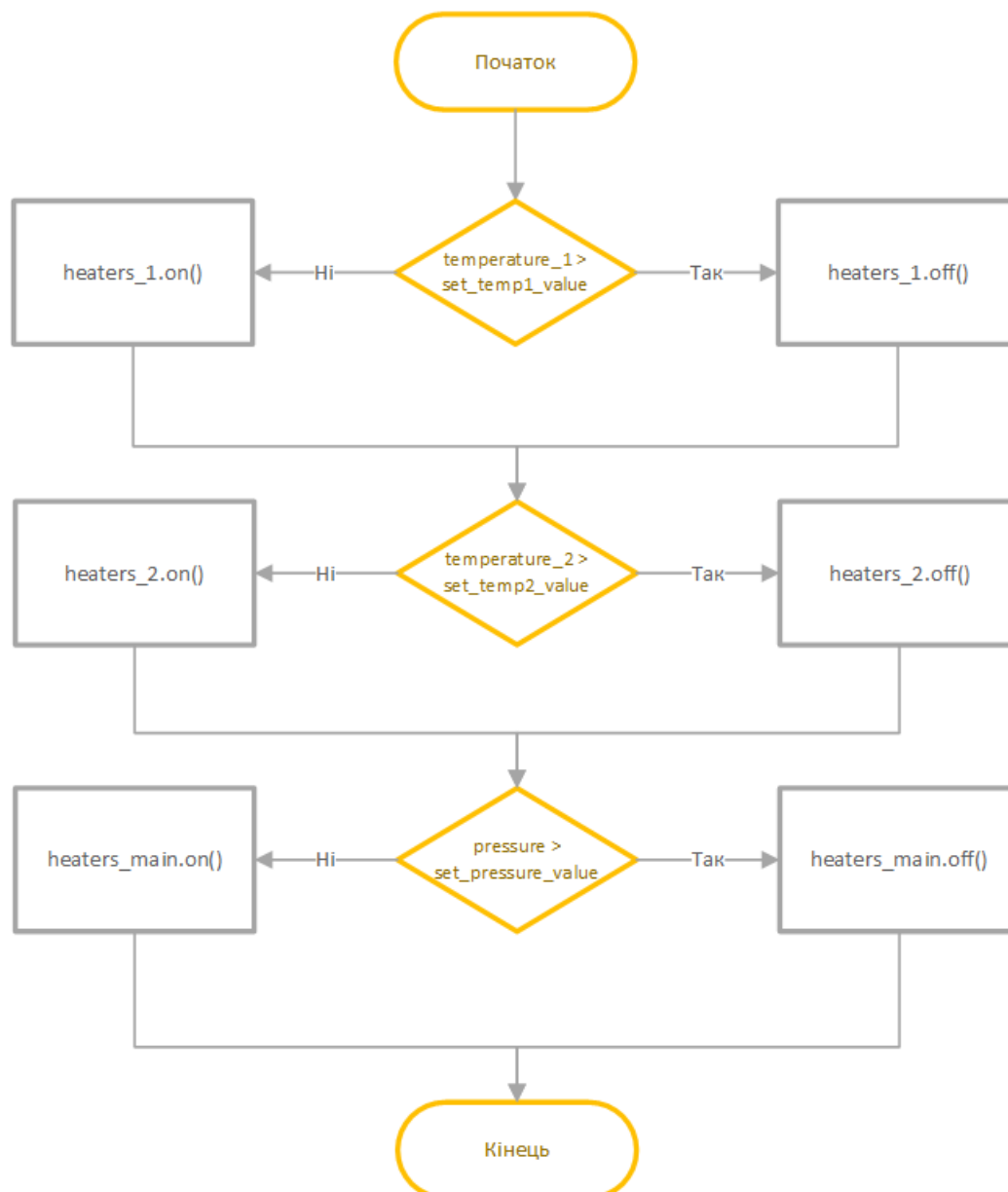


Рис. 1.22. Алгоритм для контролю заданих параметрів

temperature\_1 – змінна у якій записано значення температури води у лівому бойлері.

temperature\_2 – змінна у якій записано значення температури води у правому бойлері.

pressure – змінна у якій записано значення тиску у паровому бойлері.

set\_temp1\_value – змінна, у яку записано значення потрібної температури води у лівому бойлері. Задається у інтерфейсі користувачем (рис. 1.21)

set\_temp2\_value – змінна, у яку записано значення потрібної температури води у правому бойлері. Задається у інтерфейсі користувачем (рис. 1.21)



set\_pressure\_value – змінна, у яку записано значення потрібного тиску у паровому бойлері. Задається також у інтерфейсі користувачем (рис. 1.21).

### **Опис роботи алгоритму**

Якщо значення температури води у лівому бойлері більше, ніж задане у інтерфейсі користувача значення підтримуваної температури, то вимикаються ТЕН лівого бойлеру. Інакше ТЕН лівого вмикаються.

Якщо значення температури води у правому бойлері більше, ніж задане у інтерфейсі користувача значення підтримуваної температури, то вимикаються ТЕН правого бойлеру. Інакше ТЕН правого бойлеру вмикаються.

Якщо значення тиску у паровому бойлері більше ніж задане, у інтерфейсі користувача, значення тиску, то вимикаються ТЕНи парового бойлеру. Якщо ж значення тиску менше, ніж задане значення – ТЕНи парового бойлеру вмикаються. Для коректної роботи машини цей алгоритм повинен, якомога частіше перевіряти актуальність стану машини.

Функція, що працює за цим алгоритмом, контролює параметри смарт-системи і виконується кожну секунду.

Програмний код знаходиться у додатку А.

## **1.4. Прототип смарт-кавоварки.**

### **1.4.1. Опис конструкції та принцип дії.**

Прототип смарт-системи приготування кави (рис. 1.23).

Основою є рама, куди кріпляться вузли кавомашини. На основу кріпиться паровий бойлер та два бойлери які знаходяться на групах заварювання.

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		33

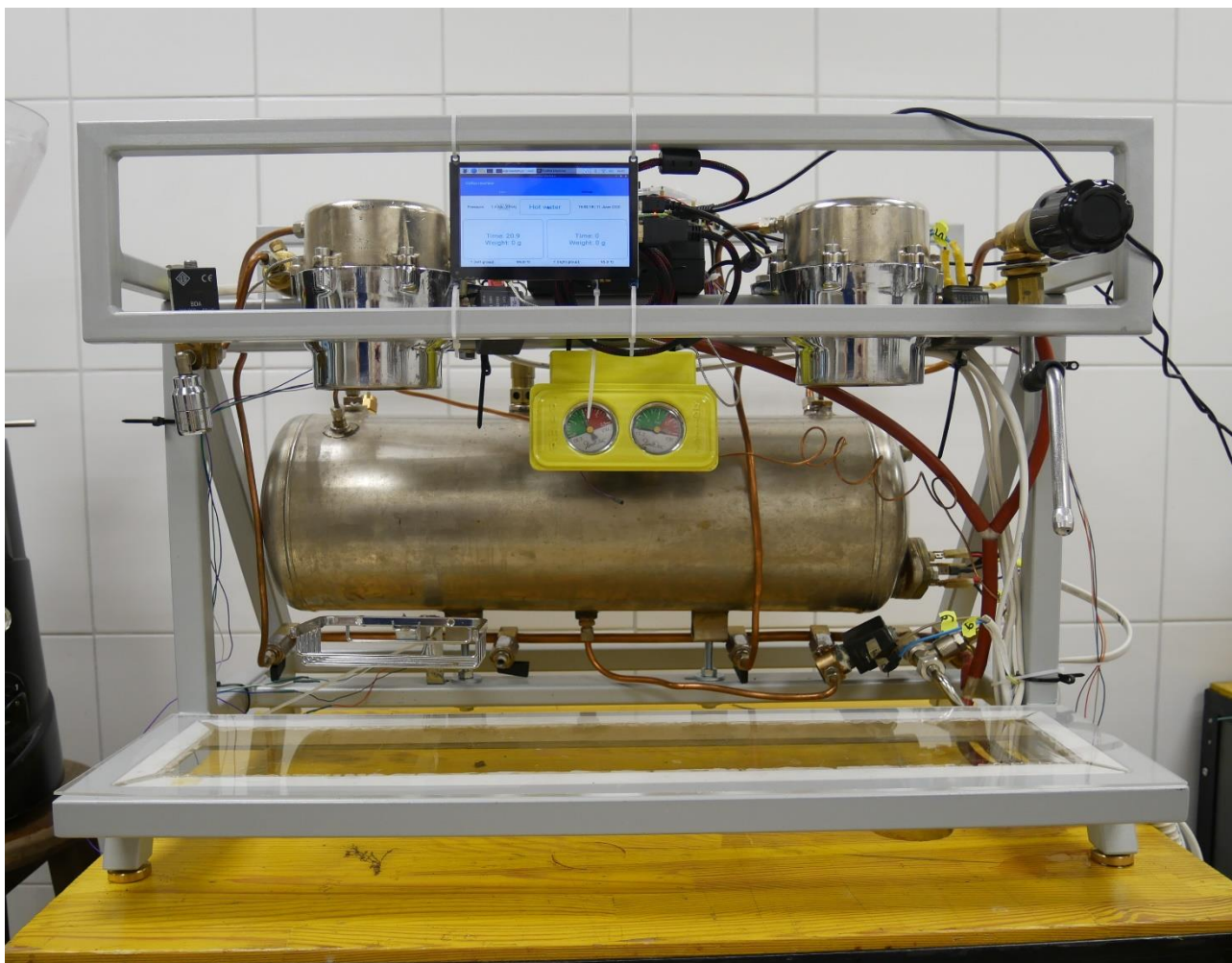


Рис. 1.23. Прототип смарт-системи приготування кави

Паровий бойлер має датчик тиску. Від значень які передаватиме датчик тиску залежить стан реле, яке замикає електричне коло і тим самим подає напругу на ТЕНи для підігріву води у паровому бойлері. На кожній групі заварювання є свій бойлер із датчиком температури. У цих бойлерах увімкнення реле залежить від значень температури води. На лицьовій стороні кожної групи є підставка, що у конструкції із тензодатчиком вимірює вагу чашки з кавою (рис. 1.5). Також прототип обладнаний сенсорним екраном, що закріплений у верхній його частині. На екран виводяться значення температури на бойлерах груп, значення тиску у паровому бойлері та вага чашок на підставках. Подача води відбувається за допомогою реле, яке вмикаючись, подає напругу на електроклапан, що в свою чергу дозволяє подачу води.

Рама для прототипу кавоварки була виготовлена на замовлення. Паровий бойлер, бойлери груп, pompa, паровий кран та електроклапани було взято із кавоварки, електроніка якої вийшла з ладу. Але всі ці запчастини запросто можна замовити у інтернеті.

Для управління електроклапанами та ТЕНами потрібна напруга 220В. Тому для цього була розроблена електрична схема (рис. 1.24).

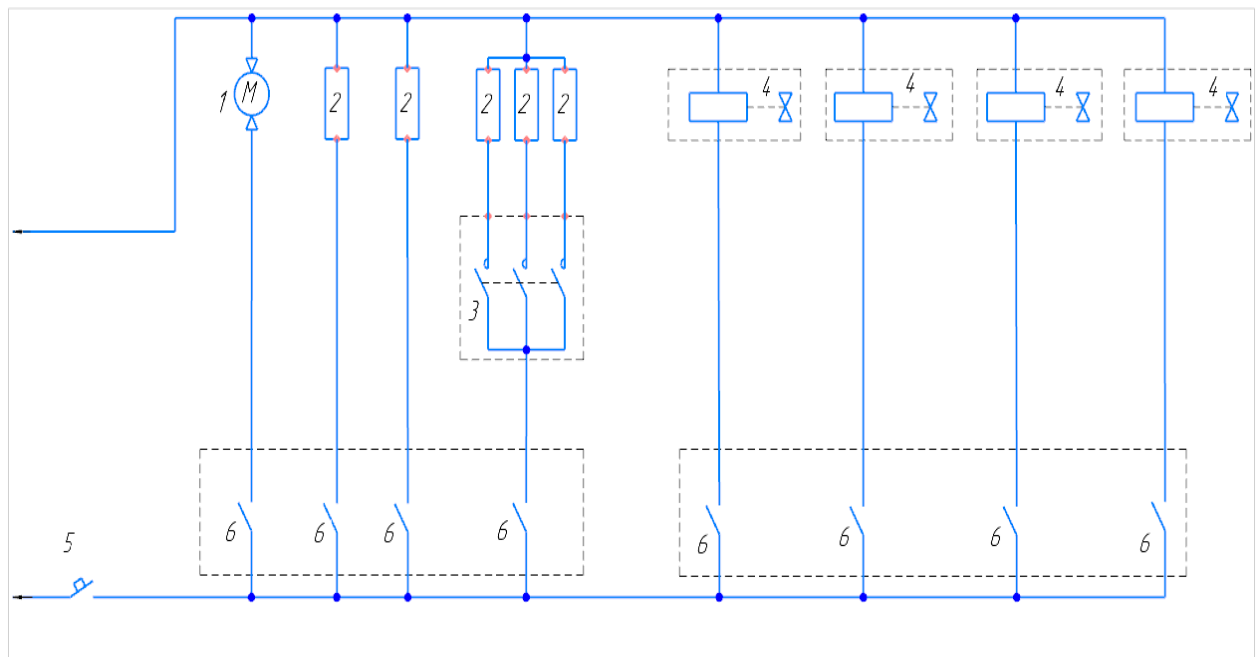


Рис. 1.24. Електрична схема

- 1 – Помпа
- 2 – ТЕН
- 3 – Контактор
- 4 – Електромагнітний клапан
- 5 – Автоматичний вимикач
- 6 – Реле

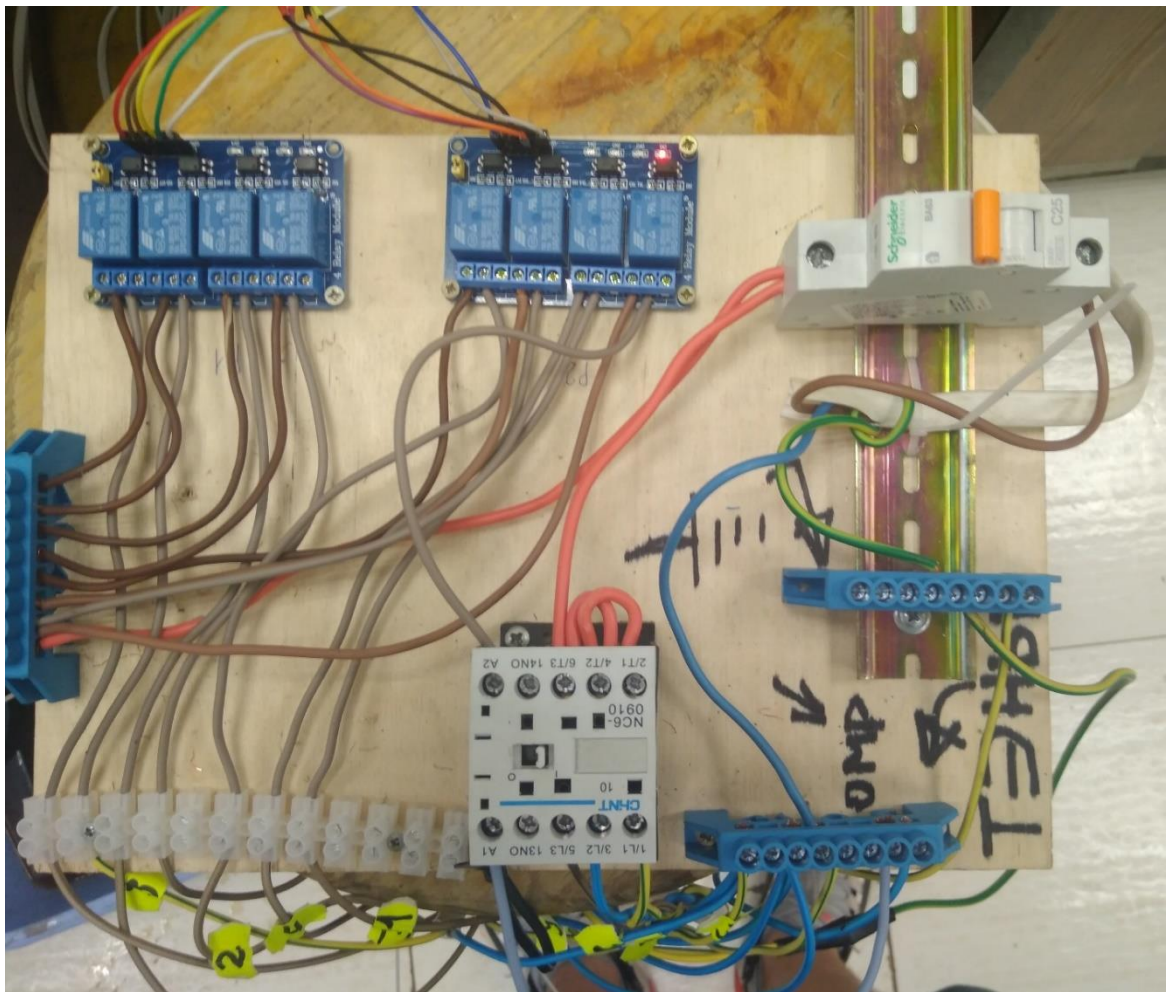


Рис. 1.25. Електричне коло для подачі напруги 220 В на ТЕНи та електроклапани

#### 1.4.2. Складові Компоненти прототипу

##### Паровий бойлер



Рис. 1.26. Паровий бойлер

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		36

Паровий бойлер – одна із основних частин кавоварки. Зазвичай його роблять із сталі або міді. Має 3 ТЕНи, які саме і гріють воду. При нагріванні води у бойлері створюється тиск, який підтримують на рівні 1.5 бар. При такому тиску температура кипіння води збільшується до 110-112 °С. У бойлері нагрівається вода, але вона не подається на групи. Вода із парового бойлеру йде на окремий кран гарячої води та на паровий кран. Бойлер оснащений запобіжним (вибуховим) клапаном який запобігає розриву корпусу.

### Бойлери груп



Рис. 1.27. Група заварювання із бойлером

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37





Рис. 1.28. Змонтована група заварювання із бойлером (вид зверху)

Бойлери груп заварювання мають по одному ТЕНу. Температура у цих бойлерах в середньому має бути 92 °С. Бойлери розташовуються на роздавальних групах, де і відбувається приготування кави.

### Помпа



Рис. 1.29. Помпа, що подає воду у бойлери із бутля

Помпа – мотор, що використовується для подачі води у бойлери. Вода у паровий бойлер подається тоді коли від Raspberry Pi поступає відповідна

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		38

команда. Паровий бойлер має бути заповнений водою на 60%. У бойлери груп заварювання вода подається при натисканні на відповідну кнопку на дисплеї управління кавоваркою.

### Крани

Паровий кран (рис. 1.30) подає сухий пар для спінювання молока.

Кран гарячої води або кран окропу (рис. 1.32). Використовується для приготування чаю.

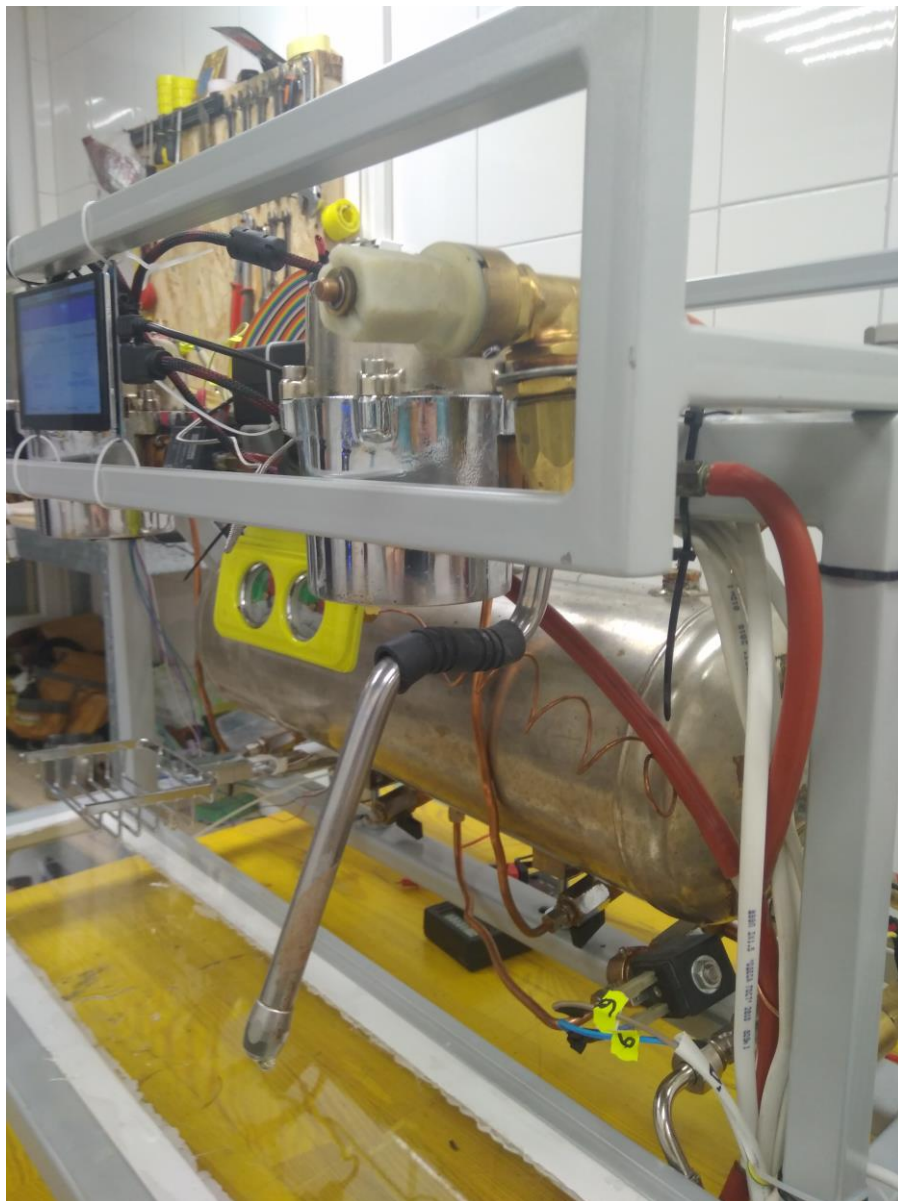


Рис. 1.30. Паровий кран

## Електромагнітний (соленоїдний) клапан

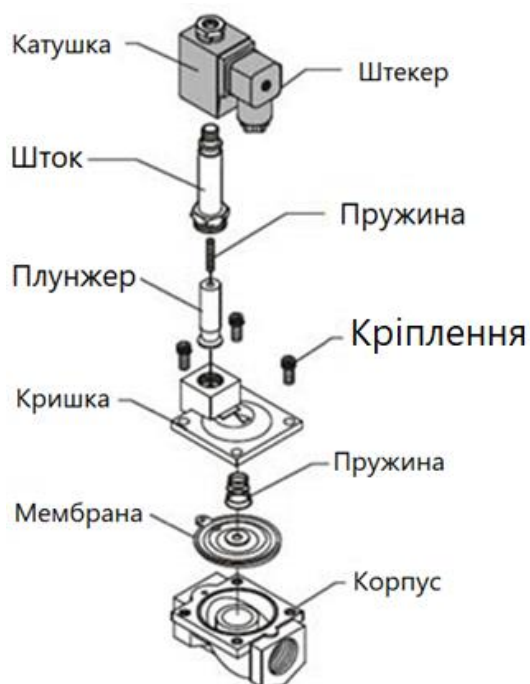


Рис. 1.31. Складові елементи електромагнітного клапану [9]

Електромагнітний клапан складається із наступних компонентів:

- Корпус. Виготовляється із нержавіючої сталі, чавуну, корозійностійкої латуні, полімерів.
- Індукційна котушка із сердечником (соленоїд). Розташована у герметичному корпусі, обмотка виконана з високоомічної технічної міді.
- Ущільнювач. Для забезпечення максимальної герметичності використовується полімер політетрафторетилен (тефлон), термостійка гума, силікон, каучук, фторопласт.
- Функціональні елементи: плунжер, пружина, шток з нержавіючої сталі.

Електромагнітний клапан має два основних компоненти: соленоїд та корпус клапана. Клапан у стані спокою, коли на котушці відсутня напруга, завжди закритий. Клапан має електромагнітну індуктивну котушку навколо залізного сердечника в центрі, яка називається плунжером.

Коли струм протікає через соленоїд, котушка створює магнітне поле. Це створює магнітне тяжіння з плунжером, переміщуючи його і долаючи силу

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		40



пружини. Плунжер піднімається так, щоб ущільнювач відкривав отвір і дозволяв протікати воді через електроклапан.

Допоки електромагнітна індуктивна котушка знаходиться під напругою – клапан буде залишатись у відкритому стані.

У даному проєкті електроклапани використовуються для подачі води у паровий бойлер, у бойлери груп та подачі води у кран.



Рис. 1.32. Електроклапан на крані гарячої води

## 2. ТЕХНОЛОГІЧНИЙ РОЗДІЛ

### 2.1. Короткий опис складальної одиниці

У якості аналізованої складальної одиниці було вибрано «мозок» смарт-системи, а саме систему управління кавомашиною, що являє собою плату Raspberry Pi із підключеними до неї датчиками та модулями.

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

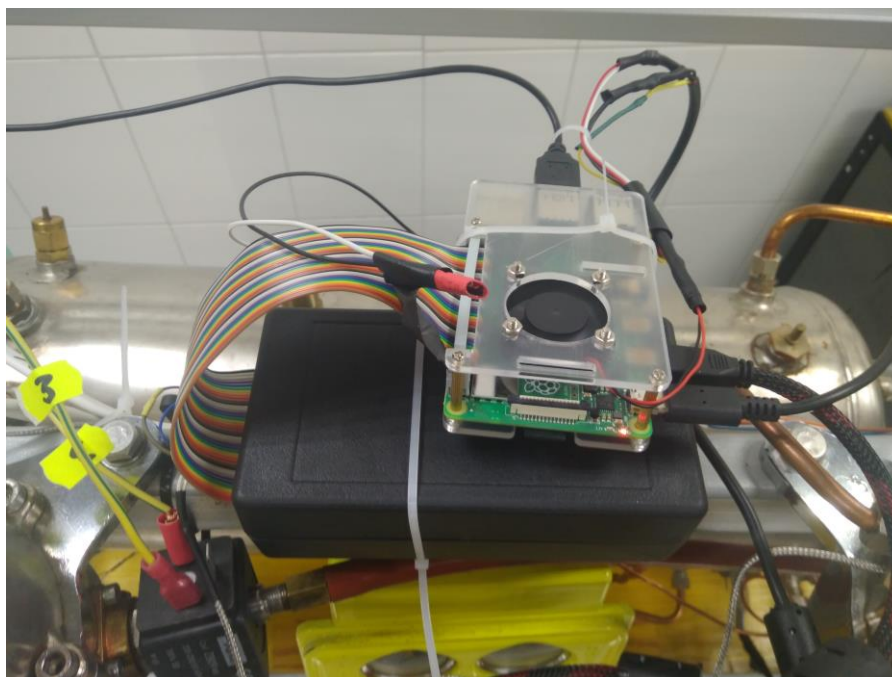


Рис. 2.1. Система управління кавомашиною

## 2.2. Визначення основних показників технологічності

Абсолютний техніко-економічний показник трудомісткості виготовлення  $T_v$  виражається сумою нормо-годин, витрачених на підготовку виробу:

$$T_v = \sum T_i, \quad (2.1)$$

де  $T_i$  – трудомісткість виготовлення та випробувань  $i$ -ї складової частини виробу в нормо-годинах.

Рівень технологічності конструкції по трудомісткості виготовлення  $K_{p.t}$  визначається як відношення досягнутої трудомісткості виробу  $T_v = 1,08$  нормо-годин до базового показника трудомісткості виготовлення  $T_{б.в} = 1,2$  нормо-годин.

$$K_{p.t} = T_v / T_{б.в} = 0,9. \quad (2.2)$$

Попередній розрахунок  $K_{p.t}$  в процесі проєктування виробу можна робити по наближеним розрахункам трудомісткості виготовлення основних складових частин, використовуючи дослідно-статистичні дані по výroбах-представниках і коригувальні коефіцієнти.

Технологічна собівартість виробу  $C_T$  визначається як сума витрат на одиницю виробу:

$$C_T = C_M + C_3 + C_{ц.в} = 6652 + 300 + 71 = 7023 \text{ грн}, \quad (2.3)$$

де  $C_M = 6652$  грн – вартість матеріалів, витрачених на виготовлення виробу;  
 $C_3 = 300$  грн – заробітна платня виробничих робітників з нарахуваннями;  
 $C_{ц.в} = 71$  грн – цехові витрати, що включають у себе витрати на електроенергію, споживану устаткуванням, на ремонт та амортизацію устаткування, інструмента і пристосувань, на мастильні, охолоджуючі, обтиральні та інші матеріали, передбачені процесом виробництва виробу.

Рівень технологічності конструкції по технологічній собівартості визначається як відношення досягнутої собівартості виробу  $C_T$  до технологічної собівартості базового виробу  $C_{б.т} = 7747$  грн

$$K_{p.c} = C_T / C_{б.т} = 0,9. \quad (2.4)$$

Попередній розрахунок  $K_{p.c}$  в процесі проектування виробу можна робити по наближеним розрахункам технологічної собівартості основних складових частин, використовуючи дослідно-статистичні дані по виробках-представниках та коригувальні коефіцієнти.

### 2.3. Визначення додаткових показників технологічності

Коефіцієнт уніфікації виробу

$$K_y = \frac{N_y + n_y}{N + n}, \quad (2.5)$$

де  $N_y = 1$  – кількість уніфікованих складальних одиниць;

$n_y = 6$  – кількість уніфікованих деталей;

$N = 1$  – кількість усіх складальних одиниць;

$n = 8$  – кількість усіх деталей

Згідно з формулою (6.5) отримуємо:

$$K_y = (1+6)/(1+8) = 0,778.$$

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

Коефіцієнт уніфікації складальних одиниць  $K_{уск}$  визначається як відношення кількості уніфікованих складальних одиниць до їхнього загального числа:

$$K_{уск} = \frac{N_y}{N} = \frac{1}{1} = 1. \quad (2.6)$$

Коефіцієнт уніфікації деталей  $K_{уд}$ :

$$K_{уд} = \frac{n_y}{n} = \frac{6}{8} = 0,75. \quad (2.7)$$

Коефіцієнт складності конструкції  $K_{ск}$  враховує ступінь розчленованості усієї конструкції виробу на складальні одиниці і деталі і визначається як

$$K_{ск} = \frac{N}{n} = \frac{1}{8} = 0,125. \quad (2.8)$$

#### 2.4. Визначення комплексного показника технологічності виробу

Комплексні показники технологічності конструкції на відміну від часткових характеризують не окремі часткові ознаки технологічності, а визначену групу ознак технологічності конструкції виробу.

Технологічність конструкції виробу може характеризуватися одним комплексним показником чи декількома, що узагальнюють групи часткових показників чи виражають різні види технологічності конструкції виробу.

Методика визначення комплексних показників враховує різну економічну ефективність вхідних часткових показників шляхом введення коефіцієнтів економічної ефективності  $K_e$  часткових показників технологічності конструкції виробу. Цей коефіцієнт доцільно обмежувати межами  $0 < K_e < 1$ .

Технологічність конструкції виробу оцінюється основними та допоміжними показниками з урахуванням коефіцієнтів економічної ефективності  $K_e$ . Коефіцієнт економічної ефективності визначаються з умов:

$$\sum_{i=1}^n K_{ei} = 1. \quad (2.9)$$

Комплексний показник технологічності розраховується за формулою:

$$K = \frac{K_1 \cdot K_{1e} + K_2 \cdot K_{2e} + K_3 \cdot K_{3e} + \dots + K_n \cdot K_{ne}}{K_{1e} + K_{2e} + K_{3e} + \dots + K_{ne}}, \quad (2.10)$$

де

$$K_1 = K_{пр} = 0,9;$$

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		44

$$K_2 = K_{pc} = 0,907;$$

$$K_3 = K_y = 0,778;$$

$$K_4 = K_{yck} = 1;$$

$$K_5 = K_{yd} = 0,75;$$

$$K_6 = K_{ck} = 0,125;$$

$$K = \frac{0,9 \cdot 0,26 + 0,907 \cdot 0,26 + 0,778 \cdot 0,12 + 1 \cdot 0,12 + 0,75 \cdot 0,12 + 0,125 \cdot 0,12}{0,26 + 0,26 + 0,12 + 0,12 + 0,12 + 0,12} = 0,788 \quad (2.11)$$

## 2.5. Схема складального складу

Схема складального складу добре ілюструє побудову складальної одиниці або приладу, отриманої на основі аналізу конструкторської документації: креслень, специфікації, комплектувальних відомостей і технічних умов. Вона є першим етапом на шляху створення технологій складання і розробляється при конструюванні приладу конструктором. Вона показує, найменування і кількість складальних одиниць приладу, число деталей, що входять до їх складу. Складальна одиниця або прилад розчленовується на окремі компоненти. Залежно від складності ці елементи розміщують за ступенями схеми складання. На нижчому, або першому, ступені схеми розміщують найпростіші, первинні складальні елементи – деталі. Вище розміщують прості отримані складанням вузли – складальні одиниці першого ступеню складання, далі їх з'єднують у складніші структури – складальні одиниці другого ступеня складання і так далі, залежно від складності елемента, аж до останнього верхнього ступеня схеми, де розміщується останній складальний елемент – виріб. Вона використовується як комплектуюча карта при складанні, але не відображує порядку складання, не вказує, з чого починати складання і в якій послідовності його вести [10].

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

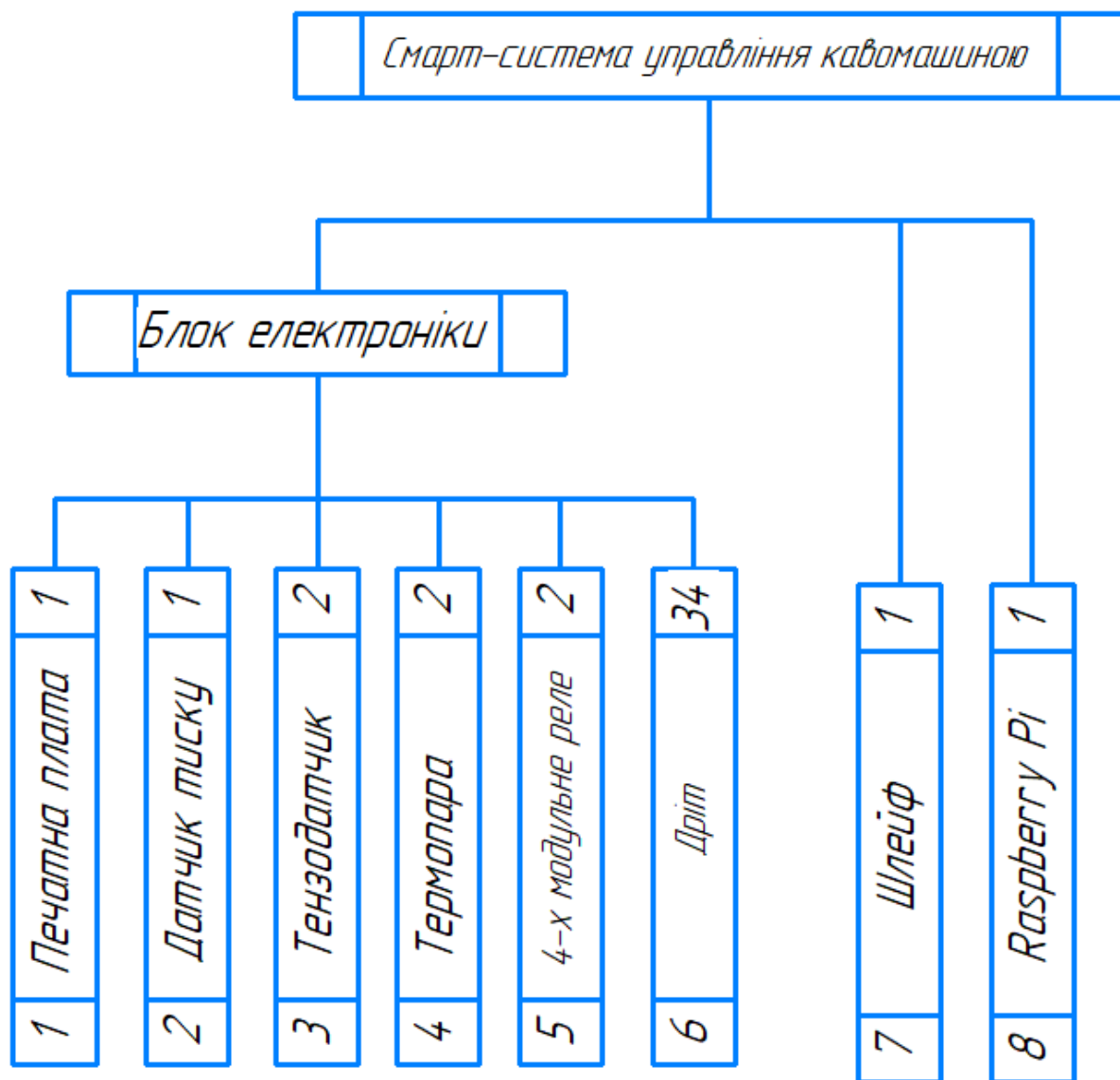


Рис. 2.2. Схема складального складу

## 2.7. Технологічна схема складання

Для повного уявлення проходження процесу складання приладу розробляється технологічна схема складання, яку розробляє технолог-складальник на початку проєктування технологічного процесу складання. Така схема наочно показує з чого починати складання, які, і в якій послідовності, деталі і складальні одиниці і в якій послідовності подавати їх на складання. Проєктування технологічної схеми складання починається з вибору базової деталі, з якої починається процес загального складання. За базову деталь приймають складну деталь на яку встановлюють всі інші деталі та вузли. Вона завжди розташовується з лівої сторони схеми. Лінія складання вказує рух до готового виробу. Складання проводиться зліва на право від базової деталі (складальної і) до готового виробу. Позначення деталей і складаних одиниць відповідає позиції вказаній на креслені і в специфікації, при цьому ліворуч вказують позицію, а праворуч – кількість деталей або складальних одиниць. Знизу на лінію загального складання подаються складальні одиниці складання, що показуються лініями складання складальних одиниць. Складання складальної одиниці також починається з базової деталі і для відмінності від загального складання в основному розташовується вертикально. При розташування базової деталі зліва, складальні одиниці та нестандартні (спеціальні) деталі подаються на лінію складання знизу, окремі стандартні деталі (деталі кріплення) а також матеріали (клей, лак, мастило тощо) подаються зверху. Зверху, також, за допомогою окремих полиць вказуються особливі види складальних робіт та контролю. Відповідно обраній послідовності подачі деталей та складальних одиниць на лінію складання за технологічною схемою складання визначають операції в технологічному процесі та проєктують технологічний процес з розподіленням його на складальні операції та переходи [10].

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		47

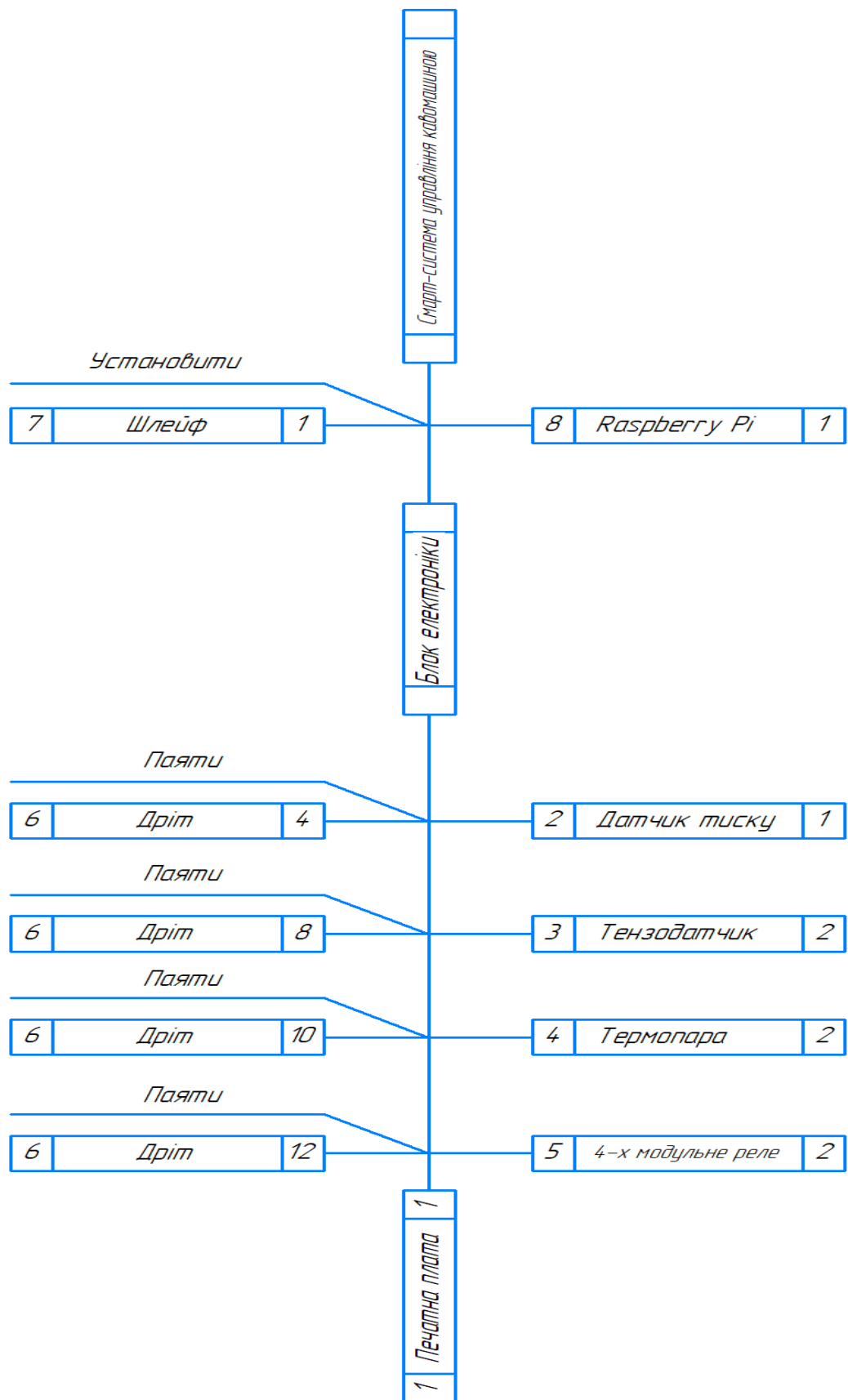


Рис. 2.3. Технологічна схема складання



## 2.8. Розмірний ланцюг

Для розрахунку розмірного ланцюга було вибрано конструкцію підставки для чашки на групі заварювання із можливістю виміру ваги за допомогою тензодатчика. Ця конструкція використовується у прототипі смарт-системи приготування кави. При її збиранні необхідно забезпечити, щоб кріплення було закріплене якомога по центру тензодатчика, бо неправильне закріплення призведе до невірних показів датчика. Для цього потрібно скласти та розрахувати схему розмірного ланцюга.

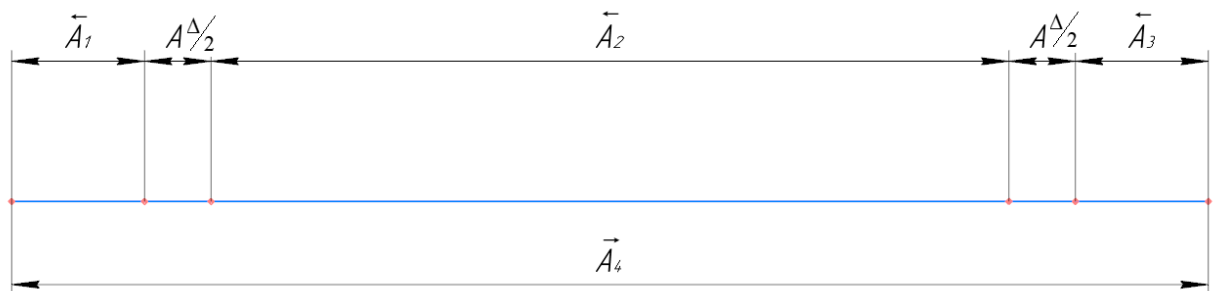


Рис. 2.4. Схема розмірного ланцюга

У таблиці приведені значення чотирьох складових ланок розмірного ланцюга та їх допустимі відхилення.

Табл. 2.1.

Ланка	Номінальний розмір	Допустиме відхилення
$A_1$	2	-0,06
$A_2$	12	0,2
$A_3$	2	-0,06
$A_4$	18	0,2

Потрібно вирішити завдання, що заключається у визначенні розміру та допуску зазору.

У розмірному ланцюгу  $\bar{A}_1$ ,  $\bar{A}_2$ ,  $\bar{A}_3$  - зменшуючі ланки,  $\bar{A}_4$  - збільшуюча ланка. Оскільки всі ланки розмірного ланцюга паралельні, то номінальний розмір замикаючої ланки  $A_\Delta$ , що є сумою двох рівних ланок  $A_{\Delta/2}$ , дорівнює:

$$A_{\Delta} = \sum_{i=1}^n A_{i36} - \sum_{i=1}^n A_{i3M} = 18 - (2 + 12 + 2) = 2 \text{ мм}; \quad (2.12)$$

відповідно

$$\frac{A_{\Delta}}{2} = 1 \text{ мм}. \quad (2.13)$$

Визначимо допуск замикаючої ланки на випадок повної взаємної замінності:

$$\sigma_{\Delta} = \sum_{i=1}^n |\sigma_i| = 0,06 + 0,2 + 0,06 + 0,2 = 0,52 \text{ мм}; \quad (2.14)$$

$$\sigma_{\Delta/2} = \frac{0,52}{2} = 0,26 \text{ мм}. \quad (2.15)$$

Координата середини поля допуску замикаючої ланки  $K_3$  обчислюється за формулою:

$$K_3 = \sum_{i=1}^n K_{i36} - \sum_{i=1}^n K_{i3M} = 0,1 - (0,1 - 0,03 - 0,03) = 0,06 \text{ мм}; \quad (2.16)$$

$$K_{3/2} = \frac{0,06}{2} = 0,03 \text{ мм}. \quad (2.17)$$

Верхнє та нижнє відхилення розміру замикаючої ланки:

$$\sigma_{\Delta B} = K_3 + 0,5 \cdot \sigma_{\Delta} = 0,06 + 0,5 \cdot 0,52 = 0,32 \text{ мм}; \quad (2.18)$$

$$\sigma_{\Delta B/2} = \frac{0,32}{2} = 0,16 \text{ мм}; \quad (2.19)$$

$$\sigma_{\Delta H} = K_3 - 0,5 \cdot \sigma_{\Delta} = 0,06 - 0,5 \cdot 0,52 = -0,2 \text{ мм}; \quad (2.20)$$

$$\sigma_{\Delta H/2} = \frac{-0,21}{2} = -0,1 \text{ мм}. \quad (2.21)$$

Таким чином, у випадку повної взаємної замінності:

$$\frac{A_{\Delta}}{2} = \left( 1 \begin{smallmatrix} +0,16 \\ -0,1 \end{smallmatrix} \right) \text{ мм}. \quad (2.22)$$

Визначимо допуск замикаючої ланки методом неповної взаємно-замінності:

$$\sigma_{\Delta} = \frac{1}{3} t_{\Delta} \sqrt{\sum_{i=1}^n (K_i)^2 \cdot (\xi_i)^2 \cdot (\sigma_i)^2}. \quad (2.23)$$

Припускаючи, що закони розподілення складових розмірів ланцюга нормальні, та з врахуванням того, що  $|\xi_i| = 1$  при вірогіднісному ризику в 0,27%,  $t_{\Delta}=3$ ,  $K_i=1$ , а також оскільки це велико-серійне виробництво, то поле розсіювання замикаючої ланки дорівнює:

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		50

$$\sigma_{\Delta} = \frac{1}{3} \cdot 3\sqrt{0,06^2 + 0,2^2 + 0,06^2 + 0,2^2} = 0,295 \text{ мм.} \quad (2.24)$$

Із врахуванням коефіцієнтів відносного розсіювання розмірів  $\alpha_i$ , значення яких для ланок розмірного ланцюга дорівнює нулю, визначимо координату середини поля розсіювання замикаючої ланки:

$$K_3 = \sum_{i=1}^n (K_{i36} + 0,5 \cdot \alpha_i \cdot \sigma_{i36}) - \sum_{i=1}^n (K_{i3M} + 0,5 \cdot \alpha_i \cdot \sigma_{i3M}) = 0,06 \text{ мм.} \quad (2.25)$$

Верхня та нижня межі поля розсіювання проміжку:

$$\sigma_{\Delta B} = K_3 + 0,5 \cdot \sigma_{\Delta} = 0,06 + 0,5 \cdot 0,295 = 0,748 \text{ мм} \quad (2.26)$$

$$\sigma_{\Delta B/2} = \frac{0,748}{2} = 0,374 \text{ мм;} \quad (2.27)$$

$$\sigma_{\Delta H} = K_3 - 0,5 \cdot \sigma_{\Delta} = 0,06 - 0,5 \cdot 0,295 = 0,453 \text{ мм;} \quad (2.28)$$

$$\sigma_{\Delta H/2} = \frac{0,453}{2} = 0,227 \text{ мм.} \quad (2.29)$$

Таким чином, при частковій взаємній заміності замикаюча ланка:

$$\frac{A_{\Delta}}{2} = \left( 1 \begin{array}{c} +0,374 \\ +0,227 \end{array} \right) \text{ мм.} \quad (2.30)$$

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		51

## ВИСНОВКИ

У результаті роботи над проєктом було проведено огляд та аналіз існуючих смарт-систем, що використовуються для приготування кави, було виконано підбір датчиків, компонентів та платформи для управління. Було розроблено програмне забезпечення для керування електричними компонентами на мові програмування Python, розроблено інтерфейс користувача за допомогою бібліотеки Kivy.

Створено робочий прототип із використанням вузлів інших існуючих кавомашин. За допомогою розробленого програмного забезпечення можна контролювати процес приготування кави.

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		52

## ПЕРЕЛІК ПОСИЛАНЬ

1. <https://scanews.coffee/2019/12/17/the-sca-publishes-updated-coffee-map-of-western-europe/>
2. <https://www.raspberrypi.org/documentation/usage/gpio/images/GPIO-Pinout-Diagram-2.png>
3. [https://arduino.ua/products\\_pictures/large\\_ADC215-3.jpg](https://arduino.ua/products_pictures/large_ADC215-3.jpg)
4. <https://github.com/tatobari/hx711py>
5. [https://arduino.ua/products\\_pictures/medium\\_ADC201-1.jpg](https://arduino.ua/products_pictures/medium_ADC201-1.jpg)
6. [https://arduino.ua/products\\_pictures/large\\_DPC355-1.jpg](https://arduino.ua/products_pictures/large_DPC355-1.jpg)
7. <https://uk.wikipedia.org/wiki/Python>
8. <https://github.com/HeaTTheatR/KivyMD>
9. <http://www.watervalve.ru/files/images/ustr1.jpg>
10. [http://old.kafvp.kpi.ua/media/metodi/tpb\\_lr\\_10.pdf](http://old.kafvp.kpi.ua/media/metodi/tpb_lr_10.pdf)

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

## ДОДАТОК А

### ЛІСТИНГ ФАЙЛІВ ПРОГРАМИ

#### Лістинг файлу raspberry.py

```
import RPi.GPIO as GPIO

# thermocouple
from MAX6675.MAX6675 import MAX6675

# weight sensor
from hx711 import HX711

import board
import busio

# i2c extender
import adafruit_tca9548a

# pressure
import adafruit_ads1x15.ads1015 as ADS
from adafruit_ads1x15.analog_in import AnalogIn

from adafruit_ina219 import ADCResolution, BusVoltageRange, INA219

import sys
import time
import traceback

GPIO.setmode(GPIO.BCM)

class Relay:
    def __init__(self, pin):
        self.pin = pin
        GPIO.setup(pin, GPIO.OUT)
        self.off()

    @property
    def state(self):
        return GPIO.input(self.pin)

    @property
    def is_on(self):
        return not self.state

    @property
    def is_off(self):
        return self.state

    def on(self):
        GPIO.output(self.pin, GPIO.LOW)
```

					ДП ПМ6115.000.000 ПЗ	Арк.
						54
Зм.	Арк.	№ докум.	Підпис	Дата		

```

def off(self):
    GPIO.output(self.pin, GPIO.HIGH)

class RaspberryPi:

    def __init__(self):
        # HX711(DT, SCK)
        self.scale1 = HX711(20, 16)
        self.scale1.set_reference_unit(1930)
        self.scale1.reset()
        self.scale1.tare()

        self.scale2 = HX711(19, 26)
        self.scale2.set_reference_unit(1)
        self.scale2.reset()
        self.scale2.tare()

        self.weight1 = 0
        self.weight2 = 0

        # MAX6675(SCK, CS, SO)
        self.thermocouple1 = MAX6675(4, 14, 15)
        self.thermocouple2 = MAX6675(17, 18, 27)

        self.relay_set_1 = {
            'K1': Relay(8),
            'K2': Relay(11),
            'K3': Relay(25),
            'K4': Relay(9),
        }

        self.relay_set_2 = {
            'K1': Relay(22),
            'K2': Relay(23),
            'K3': Relay(24),
            'K4': Relay(10),
        }

        i2c = busio.I2C(board.SCL, board.SDA)
        tca = adafruit_tca9548a.TCA9548A(i2c)

        ads = ADS.ADS1015(tca[0])
        self.pressure_chan = AnalogIn(ads, ADS.P0)

    def get_pressure(self):
        return self.pressure_chan.value

    def make_weight1(self):
        weight = max(0, int(self.scale1.get_weight(1)))
        self.weight1 = weight

```

					ДП ПМ6115.000.000 ПЗ	Арк.
						55
Зм.	Арк.	№ докум.	Підпис	Дата		

```

        self.scale1.power_down()
        self.scale1.power_up()

    def make_weight2(self):
        weight = max(0, int(self.scale2.get_weight(1)))
        self.weight2 = weight
        self.scale2.power_down()
        self.scale2.power_up()

    @staticmethod
    def cleanup():
        GPIO.cleanup()

```

### Лістинг файлу services.py

```

from raspberry import RaspberryPi

class CoffeeMachineService:
    """Клас для управління кавомашиною"""
    set_pressure_value = 6400
    set_temp1_value = 50
    set_temp2_value = 50

    def __init__(self):
        self.raspberry = RaspberryPi()
        # Мотор для води
        self.engine_for_water = self.raspberry.relay_set_1['K1']
        # Електроклапан головного бойлеру
        self.valve_main_boiler = self.raspberry.relay_set_1['K2']
        # Електроклапан лівого бойлеру
        self.valve_1_boiler = self.raspberry.relay_set_1['K3']
        # Електроклапан правого бойлеру
        self.valve_2_boiler = self.raspberry.relay_set_1['K4']
        # ТЕНи головного бойлеру
        self.heaters_main = self.raspberry.relay_set_2['K1']
        # ТЕНи лівого бойлеру
        self.heaters_1 = self.raspberry.relay_set_2['K2']
        # ТЕНи правого бойлеру
        self.heaters_2 = self.raspberry.relay_set_2['K3']
        # Електроклапан для крану гарячої води
        self.hot_water_valve = self.raspberry.relay_set_2['K4']

    def engine_off(self):
        """Виключення мотору із перевіркою чи не використовується
        він іншими вузлами"""
        if self.valve_main_boiler.is_off and
        self.valve_1_boiler.is_off and self.valve_2_boiler.is_off:
            self.engine_for_water.off()

    def engine_on(self):

```

					ДП ПМ6115.000.000 ПЗ	Арк.
						56
Зм.	Арк.	№ докум.	Підпис	Дата		



```

        self.engine_for_water.on()

@property
def temperature_1(self):
    return self.raspberry.thermocouple1.readTempC()

@property
def temperature_2(self):
    return self.raspberry.thermocouple2.readTempC()

@property
def pressure(self):
    return self.raspberry.get_pressure()

def set_pressure(self, value):
    self.set_pressure_value = value

def set_temperature_1(self, value):
    self.set_temp1_value = value

def set_temperature_2(self, value):
    self.set_temp2_value = value

def boilers_critical_parameters_control(self, dt=None):
    if self.temperature_1 > 96:
        self.heaters_1.off()
    elif self.temperature_2 < 87:
        self.heaters_1.on()

    if self.temperature_2 > 96:
        self.heaters_2.off()
    elif self.temperature_2 < 87:
        self.heaters_2.on()

    if self.pressure > 7200:
        self.heaters_main.off()
    elif self.pressure < 6000:
        self.heaters_main.on()

def boilers_parameters_control(self, dt=None):
    """ Функція для контролю заданих параметрів """
    if self.temperature_1 > self.set_temp1_value:
        self.heaters_1.off()
    elif self.temperature_2 < self.set_temp1_value:
        self.heaters_1.on()

    if self.temperature_2 > self.set_temp2_value:
        self.heaters_2.off()
    elif self.temperature_2 < self.set_temp2_value:
        self.heaters_2.on()

    if self.pressure > self.set_pressure_value:

```

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

```

        self.heaters_main.off()
    elif self.pressure < self.set_pressure_value:
        self.heaters_main.on()

def turn_off(self):
    self.raspberry.cleanup()

```

### Лістинг файлу utils.py

```

import time

class Timer:
    start_time = 0
    end_time = 0

    @property
    def duration(self):
        if self.started:
            value = time.time()
            if self.ended:
                value = self.end_time
            return round(value - self.start_time, 1)
        return 0

    def start(self):
        self.start_time = time.time()

    def end(self):
        self.end_time = time.time()

    @property
    def started(self):
        if self.start_time:
            return True
        return False

    @property
    def ended(self):
        if self.end_time:
            return True
        return False

    def nullify(self):
        self.start_time = 0
        self.end_time = 0

```

### Лістинг файлу main.py

```

from kivy.config import Config
from kivy.core.window import Window

```

					ДП ПМ6115.000.000 ПЗ	Арк.
						58
Зм.	Арк.	№ докум.	Підпис	Дата		

```

from kivy.uix.gridlayout import GridLayout
from kivy.uix.boxlayout import BoxLayout
from kivy.clock import Clock

from kivymd.app import MDApp
from kivymd.uix.toolbar import MDToolbar
from kivymd.uix.tab import MDTabsBase, MDTabs

from services import CoffeeMachineService
from utils import Timer

import time

coffee_machine = CoffeeMachineService()

class MainContainer(GridLayout, MDTabsBase):
    weight1 = 0
    weight2 = 0
    timer1 = Timer()
    timer2 = Timer()

    def clock(self, dt):
        self.label_clock.text = time.strftime("%H:%M:%S | %d %B
%Y")

    def hot_water(self):
        """Вкл/викл крану гарячої води"""
        if coffee_machine.hot_water_valve.is_off:
            coffee_machine.hot_water_valve.on()
        else:
            coffee_machine.hot_water_valve.off()

    def group1(self):
        """Керування лівою групою заварювання"""
        if self.timer1.ended:
            self.timer1.nullify()

        if self.timer1.started:
            self.timer1.end()
            coffee_machine.valve_1_boiler.off()
            coffee_machine.engine_off()
        else:
            self.timer1.start()
            coffee_machine.valve_1_boiler.on()
            coffee_machine.engine_on()

    def group2(self):
        """Керування правою групою заварювання"""
        if self.timer2.ended:
            self.timer2.nullify()

```

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		59

```

if self.timer2.started:
    self.timer2.end()
    coffee_machine.valve_2_boiler.off()
    coffee_machine.engine_off()
else:
    self.timer2.start()
    coffee_machine.valve_2_boiler.on()
    coffee_machine.engine_on()

def get_weight1(self, dt):
    """Отримання показників вагів лівої групи заварювання"""
    coffee_machine.raspberry.make_weight1()
    self.weight1 = coffee_machine.raspberry.weight1

def get_weight2(self, dt):
    """Отримання показників вагів правої групи заварювання"""
    coffee_machine.raspberry.make_weight2()
    self.weight2 = coffee_machine.raspberry.weight2

def time1(self, dt):
    """Виведення значень ваги та таймеру лівої групи
заварювання"""
    self.button_1.text = f"Time: {self.timer1.duration}\n" \
        f"Weight: {self.weight1} g"

def time2(self, dt):
    """Виведення значень ваги та таймеру правої групи
заварювання"""
    self.button_2.text = f"Time: {self.timer2.duration}\n" \
        f"Weight: {self.weight2} g"

def machine_sensors_data(self, dt):
    """Отримання та виведення на екран значення тиску у
основному бойлері та значень
теператур у лівій та правій групі заварювання"""

    pressure = coffee_machine.pressure
    temp1 = coffee_machine.temperature_1
    temp2 = coffee_machine.temperature_2
    pressure_bar = round((pressure - 4400) * 0.00055, 2)
    self.label_pressure.text = str(pressure_bar) + " Bar (" +
str(pressure) + ")"
    self.label_temp1.text = str(temp1) + " °C"
    self.label_temp2.text = str(temp2) + " °C"

class SettingsContainer(GridLayout, MDTabsBase):

    def setup(self):
        self.pressure.text =
str(coffee_machine.set_pressure_value)
        self.temp1.text = str(coffee_machine.set_temp1_value)

```

					ДП ПМ6115.000.000 ПЗ	Арк.
						60
Зм.	Арк.	№ докум.	Підпис	Дата		

```

        self.temp2.text = str(coffee_machine.set_temp2_value)

def main_boiler(self):
    if coffee_machine.valve_main_boiler.is_off:
        coffee_machine.valve_main_boiler.on()
        coffee_machine.engine_on()
    else:
        coffee_machine.valve_main_boiler.off()
        coffee_machine.engine_off()

def set_pressure(self):
    value = self.pressure.text
    if value:
        value = float(value)
        print(f"Set pressure: {value}")
        coffee_machine.set_pressure(value)

def set_temp1(self):
    value = self.temp1.text
    if value:
        value = float(value)
        if 96 >= value:
            print(f"Set temp1: {value}")
            coffee_machine.set_temperature_1(value)

def set_temp2(self):
    value = self.temp2.text
    if value:
        value = float(value)
        if 96 >= value:
            print(f"Set temp2: {value}")
            coffee_machine.set_temperature_2(value)

class CoffeeApp(MDApp):
    title = "Coffee Machine"
    icon = "coffee.png"

    def __init__(self, **kwargs):
        self.theme_cls.theme_style = "Light"
        super().__init__(**kwargs)

    def on_request_close(self, *args):
        coffee_machine.turn_off()
        self.stop()

    def build(self):
        Window.bind(on_request_close=self.on_request_close)
        Config.set('kivy', 'keyboard_mode', 'systemandmulti')
        Config.write()

        parent = BoxLayout(orientation="vertical")

```

					ДП ПМ6115.000.000 ПЗ	Арк.
						61
Зм.	Арк.	№ докум.	Підпис	Дата		

```

parent.add_widget(MDToolbar(title="Coffee machine"))

tabs = MDTabs()
parent.add_widget(tabs)

main_container = MainContainer(text="Main")
settings_container = SettingsContainer(text="Settings")
settings_container.setup()

Clock.schedule_interval(coffee_machine.boilers_parameters_control,
1)

Clock.schedule_interval(main_container.machine_sensors_data, 1)
Clock.schedule_interval(main_container.time1, 0.1)
Clock.schedule_interval(main_container.time2, 0.1)
Clock.schedule_interval(main_container.get_weight1, 1)
Clock.schedule_interval(main_container.get_weight2, 1)
Clock.schedule_interval(main_container.clock, 1)

tabs.add_widget(main_container)
tabs.add_widget(settings_container)

return parent

if __name__ == '__main__':
    CoffeeApp().run()

```

### Лістинг файлу coffee.kv

```

<CustomLabel@MDLabel>:
    font_size: 20
    text_size: self.size
    halign: 'center'
    valign: 'center'

<LargeButton@MDRoundFlatButton>:
    font_size: 36
    text_size: self.size
    halign: 'center'
    valign: 'center'

<DefaultButton@MDRectangleFlatButton>:
    size_hint: 1, 0.9
    font_size: 24
    text_size: self.size
    halign: 'center'
    valign: 'center'

```

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		62

```

<SettingsContainer>:
  pressure: pressure
  temp1: temp1
  temp2: temp2

  btn_pressure: btn_pressure
  btn_temp1: btn_temp1
  btn_temp2: btn_temp2

  btn_main_boiler: btn_main_boiler

  rows: 4
  size_hint: 0.5, 1
  padding: 30

  BoxLayout:
    spacing: 20
    MDTextField:
      id: pressure
      input_type: 'number'
      input_filter: 'float'
      hint_text: "Pressure"
      font_size: 36
    DefaultButton:
      text: "Set"
      id: btn_pressure
      on_release:
        root.set_pressure()

  BoxLayout:
    spacing: 20
    MDTextField:
      id: temp1
      input_type: 'number'
      input_filter: 'float'
      hint_text: "Temperature 1st boiler"
      font_size: 36
    DefaultButton:
      text: "Set"
      id: btn_temp1
      on_release:
        root.set_temp1()

  BoxLayout:
    spacing: 20
    MDTextField:
      id: temp2
      input_type: 'number'
      input_filter: 'float'
      hint_text: "Temperature 2nd boiler"
      font_size: 36
    DefaultButton:
      text: "Set"
      id: btn_temp2

```

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

```

        on_release:
            root.set_temp2()
BoxLayout:
    spacing: 20
    DefaultButton:
        text: "Valve main boiler"
        id: btn_main_boiler
        on_release:
            root.main_boiler()

<MainContainer>:
    label_clock: label_clock
    label_pressure: label_pressure
    label_temp1: label_temp1
    label_temp2: label_temp2

    button_1: button_1
    button_2: button_2
    button_3: button_3

    rows: 2
    BoxLayout:
        padding: 10
        size_hint: 1, 0.45
        BoxLayout:
            CustomLabel:
                text: "Pressure: "
            CustomLabel:
                id: label_pressure
                text: ""

        BoxLayout:
            orientation: 'vertical'
            padding: 10
            size_hint: 1, 1
            LargeButton:
                size_hint: 1, 1
                id: button_3
                text: "Hot water"
                on_release:
                    root.hot_water()

        BoxLayout:
            CustomLabel:
                id: label_clock
                text: ""

    GridLayout:
        cols: 2
        BoxLayout:

```

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		64



```

orientation: 'vertical'
padding: 10
size_hint: 0.5, 1
LargeButton:
    size_hint: 1, 1
    id: button_1
    text: "1 group"
    on_release:
        root.group1()

BoxLayout:
    size_hint: 1, 0.15
    CustomLabel:
        text: "T (left group): "
    CustomLabel:
        id: label_temp1
        text: ""

BoxLayout:
    orientation: 'vertical'
    padding: 10
    size_hint: 0.5, 1
    LargeButton:
        size_hint: 1, 1
        id: button_2
        text: "2 group"
        on_release:
            root.group2()

BoxLayout:
    size_hint: 1, 0.15
    CustomLabel:
        text: "T (right group): "
    CustomLabel:
        id: label_temp2
        text: ""

```

					ДП ПМ6115.000.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		65